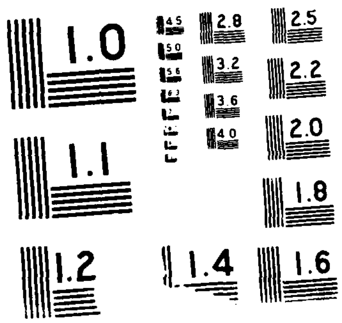
AD-A194 357 SDI (STRATEGIC DEFENSE INITIATIVE) BATTLE MANAGEMENT/C3 1/2
(COMMAND CONTROLAA (U) INSTITUTE FOR DEFENSE ANALYSES
ALEXANDRIA VA G FRENKEL ET AL APR 88 IDA-P-2068
UNCLASSIFIED IDA/HQ-87-32185 MDA903-84-C-0031 F/G 15/3 1 NL


1.0
1.1
1.2
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

(2)

IDA PAPER P-2068

SDI BATTLE MANAGEMENT/C^3 ALGORITHMS TECHNOLOGY PROGRAM PLAN

Gabriel Frenkel
Thomas S. Paterson
Maile E. Smith

April 1988

Prepared for
Strategic Defense Initiative Organization

88

IDA

INSTITUTE FOR DEFENSE ANALYSES
1801 N. Beauregard Street, Alexandria, Virginia 22311

IDA Log No. HQ 87-32185

DEFINITIONS

IDA publishes the following documents to report the results of its work.

Reports

Reports are the most authoritative and most carefully considered products IDA publishes. They normally embody results of major projects which (a) have a direct bearing on decisions affecting major programs, or (b) address issues of significant concern to the Executive Branch, the Congress and/or the public, or (c) address issues that have significant economic implications. IDA Reports are reviewed by outside panels of experts to ensure their high quality and relevance to the problems studied, and they are released by the President of IDA.

Papers

Papers normally address relatively restricted technical or policy issues. They communicate the results of special analyses, interim reports or phases of a task, ad hoc or quick reaction work. Papers are reviewed to ensure that they meet standards similar to those expected of refereed papers in professional journals.

Memorandum Reports

IDA Memorandum Reports are used for the convenience of the sponsors or the analysts to record substantive work done in quick reaction studies and major interactive technical support activities; to make available preliminary and tentative results of analyses or of working group and panel activities; to forward information that is essentially unanalyzed and unevaluated; or to make a record of conferences, meetings, or briefings, or of data developed in the course of an investigation. Review of Memorandum Reports is suited to their content and intended use.

The results of IDA work are also conveyed by briefings and informal memoranda to sponsors and others designated by the sponsors, when appropriate.

The work reported in this document was conducted under contract MDA 903 84 C 0031 for the Department of Defense. The publication of this IDA document does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official position of that agency.

This paper has been reviewed by IDA to assure that it meets high standards of thoroughness, objectivity, and sound analytical methodology and that the conclusions stem from the methodology.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A194357

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY N/A 2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
4. PERFORMING ORGANIZATION REPORT NUMBER(S) IDA Paper P-2068	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Institute for Defense Analyses 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION DoD-IDA Management Office, OUSD(A)
6c. ADDRESS (City, State, and Zip Code) 1801 N. Beauregard Street Alexandria, VA 22311	7b. ADDRESS (CITY, STATE, AND ZIP CODE) 1801 N. Beauregard Street Alexandria, VA 22311
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Strategic Defense Initiative Organization 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER MDA 903 84 C 0031
8c. ADDRESS (City, State, and Zip Code) The Pentagon Washington, DC 20301-7100	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT: — PROJECT NO.: — TASK NO.: T-R2-261-3a WORK UNIT ACCESSION NO.: —

11. TITLE (Include Security Classification)
SDI Battle Management/C^3 Algorithm Technology Program Plan

12. PERSONAL AUTHOR(S).
Gabriel Frenkel, Thomas S. Paterson, Maile E. Smith

13. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM 3/87 TO 11/87	April 1988	95

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
SDI; Algorithms; Battle Management; Command, Control, Communications

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The Institute for Defense Analyses (IDA) has collected and analyzed information on battle management algorithm technology that is relevant to Battle Management/Command, Control and Communications (BM/C^3). This Memorandum Report represents a program plan that will provide the BM/C^3 Directorate of the Strategic Defense Initiative Organization (SDIO) with administrative and technical insight into algorithm technology.

This program plan focuses on current activity in algorithm development and provides information and analysis to the SDIO to be used in formulating budget requirements for FY 1988 and beyond. Based upon analysis of algorithm requirements and ongoing programs, recommendations have been made for research areas that should be pursued, including both the continuation of current work and the initiation of new tasks.

This final report includes all relevant material from interim reports as well as new results.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Gabriel Frenkel	22b. TELEPHONE (Include Area Code) (703) 578-2829 22c. OFFICE SYMBOL

DD FORM 1473. 84 MAR

83 APR edition may be used until exhausted.
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

IDA PAPER P-2068

SDI BATTLE MANAGEMENT/C^3 ALGORITHMS TECHNOLOGY PROGRAM PLAN

Gabriel Frenkel
Thomas S. Paterson
Maile E. Smith

April 1988

INSTITUTE FOR DEFENSE ANALYSES

Contract MDA 903 84 C 0031
Task T-R2-261-3a

PREFACE

The purpose of this Memorandum Report is to provide information and analyses to the SDIO Battle Management/Command, Control and Communications (BM/C^3) Directorate in the development of a program plan for battle management algorithm technology.

An interim report was submitted earlier. This report contains all the previous material as well as new results. It has been written in parallel with a separate report on BM/C^3 networking technology.

The cutoff date for the work in this report is November 1987.

ABSTRACT

The Institute for Defense Analyses (IDA) has collected and analyzed information on battle management algorithm technology that is relevant to Battle Management/Command, Control and Communications (BM/C^3). This Memorandum Report represents a program plan that will provide the BM/C^3 Directorate of the Strategic Defense Initiative Organization (SDIO) with administrative and technical insight into algorithm technology.

This program plan focuses on current activity in algorithm development and provides information and analysis to the SDIO to be used in formulating budget requirements for FY 1988 and beyond. Based upon analysis of algorithm requirements and ongoing programs, recommendations have been made for research areas that should be pursued, including both the continuation of current work and the initiation of new tasks.

This final report includes all relevant material from interim reports as well as new results.

CONTENTS

EXECUTIVE SUMMARY

A. OVERVIEW

Battle-management algorithms are the mathematical/logical processes and procedures used to allocate resources, manage and form the files, and execute command and control actions. Whether autonomous or man-in-the-loop, these procedures are designed to operate in a robust manner responsive to change and evolving technology. Software implementation of battle-management algorithm in a loosely coupled, widely dispersed, real-time, heterogeneous multiprocessor environment is an aspect of algorithms implementation.

The purpose of this document is to provide information and analysis to the SDIO to be used in formulating budget requirements for FY 1988 and beyond. The emphasis here *is on the SDI-sponsored program, conducted through the services, but some other relevant* government-sponsored and commercial efforts also have been examined. Some of the design concepts being developed in such programs may be applicable to SDI. If so, the SDIO can avoid duplicating this work and concentrate its resources on areas critical to SDI that are not being addressed elsewhere. The survey of non-SDIO efforts was by no means exhausted, and further work in this area will continue.

This study was carried out in five major steps:

1. Define algorithm requirements.
2. Identify and analyze tasks currently funded by SDIO and other related activities.
3. Collect data about the algorithms being developed, analyze it, and highlight by features.
4. Identify programs not currently funded by SDIO BM/C^3 or others but which may contribute to the algorithm development effort for SDI BM/C^3.
5. Provide recommendations for structuring the BM/C^3 algorithm development program.

B. KEY BM/C^3 ALGORITHMS

During the review of algorithm development activity, a natural subdivision of the field of activity emerged. There were variations between agencies or contractors, but the following breakdown reflects the bulk of the development efforts ongoing at this time.

- ***Resource basing,*** which is concerned with the deployment of ground-based interceptors for point and area defense.
- ***Track initiation and maintenance***, which pertains to establishing, and by implication, predicting the tracks of threat objects from sensor observations.
- ***Discrimination*** pertains to classifying the threat objects (usually after ejection from a post-boost vehicle) into reentry vehicles (RVs) and decoys of different types.
- ***Common reference frame*** pertains to the establishment of a common spatial and temporal coordinate system between platforms to synchronize operations and transfer positional data.
- ***Threat assessment*** involves the estimation of relevant parameters of a threat as it develops.
- ***Weapon-to-target assignment*** involves the assignment of weapons to threat objects.
- ***Mode selection*** involves selection of the overall strategy most appropriate to the threat.
- ***Engagement assessment*** involves evaluation of the results of engagements as they develop.
- ***Sensor resource management.*** The allocation of the overall tasks between the sensors to achieve efficient operation.

Figure 1 (page 6) illustrates the functional interdependence, inputs, and outputs of the nine categories of algorithms shown in the nine squares with rounded corners.

C. OVERVIEW OF ALGORITHM DEVELOPMENT ACTIVITY

Figure S-1 shows the BM/C^3 algorithms identified and reviewed in this report. Not shown are the algorithms developed in the various models used for the evaluation of candidate architectures. These were found not to contain any innovative algorithms. A list of those reviewed is included in Chapter IV. Three program elements, namely the Boost Surveillance and Tracking Sensors (BSTS). the Space Surveillance and Tracking System (SSTS), and the Space-Based Interceptor (SBI) Program, have developed algorithms.

SPONSOR	CONTRACTOR	RESOURCE BASING	TRACK INITIATION AND MAINTENANCE	DISCRIMINATION	COMMON REFERENCE FRAME	SENSOR RESOURCE MANAGEMENT	THREAT ASSESSMENT	MODE SELECTION	WEAPON-TO-TARGET ASSIGNMENT	ENGAGEMENT ASSESSMENT
ARMY	ALPHATECH								●	
	ALPHATECH / HONEYWELL		●							
	NICHOLS RESEARCH CORP.		●	●						
	SPARTA	●					●	●		●
	UNISYS					●				
AIR FORCE	ALPHATECH		●						●	
	AEROSPACE			●						
	APPLIED TECH. ASSOCIATES		●							
	LOGICON								●	
	HUGHES		●							
	PAR TECHNOLOGIES		●							
NAVY	COMMAND SYSTEMS GROUP			●						
	VERAC		●							
SDIO	IDA	●		●					●	
	LOGICON								●	
	MARTIN MARIETTA								●	
	TRW								●	
IN-HOUSE	HUGHES								●	
	MITRE		●							

Figure S-1. Summary of Algorithms Studied.

In the first case, information was denied on the basis of lack of need to know--this is discussed further under conclusions and recommendations. In the case of the SSTS, there was not enough time for a thorough investigation. The SDI Project Office arranged a briefing for the IDA representatives, and provided relevant reports.

The activities listed in the table break down into four categories: those sponsored by the SDIO BM/C^3 office through the services, those sponsored directly, and two in-house activities. Of these, by far the most comprehensive are the programs of the services, since they aim at an integrated set of algorithms, each for its own testbed. The programs are described in Chapter IV, and each of the algorithms is discussed in detail in Chapter III.

D. EVALUATION OF THE STATUS OF ALGORITHM DEVELOPMENT

The status of algorithm development in the nine categories identified above is quite uneven. In the case of some types of algorithms there is considerable activity, while other vital functions are not covered. In the latter case, it is often tacitly assumed by the designer that the function is performed perfectly, which is never the case. A brief summary of the status of development for each function follows.

1. Resource Basing

Analyses of ground-based terminal defense problems involving abstract mathematical constructs, e.g., minimax problems requiring linear programming techniques, abound. They are characterized mostly by an air of unreality and disregard of important real-life factors. Some important aspects of the problem requiring further study are the following:

- The geographic dispersion of the targets to be defended, and inclusion of their value or category (silos, cities, etc.)
- The dynamic characteristic of interceptors and their shrinking footprint with decreasing commit altitude
- The impact of time-varying discrimination quality for various pen-aids
- The random nature of the threat dispersion because of attrition in previous encounters, e.g., boost-phase.

Aside from the problems discussed above, the terminal-phase battle manager faces unique problems because this is the last line of defense. The interceptor allocator must constantly consider trade-offs of current shots at somewhat credible targets against future shots at RVs in later waves. This problem has yet to be adequately addressed. Virtually every model which attempts to solve this problem makes the assumption of separate aimpoints which cannot cross-defend each other, or if cross defense is allowed, it is not done with a mission in mind, but in a subtractive fashion, where the goal is simply to maximize the number of RVs killed.

2. Track Initiation and Maintenance

A variety of algorithms are being developed for the multi-target tracking function of SDI. Some contractors have chosen to examine the problem a step at a time, slowly increasing the fidelity--and therefore complexity--of the algorithm (Mitre, ATA) while others have chosen to solve the full range of issues discussed above at once (Alphatech, Hughes). While all these algorithms claim some level of performance, and all of the contractors we talked with agree that most of the issues above are critical and need to be addressed, we have yet to see any of the sensitivity studies necessary to compare two algorithms or even to evaluate a single algorithm. Until detailed studies using, for example, Monte Carlo technniques and including error sources are available, the quality and timeliness of tracking information necessary for the strategic defense mission cannot be assured, and the effects of realistic tracking errors on the rest of battle management will remain unknown. The performance of these algorithms as a function of sensor (BSTS or SSTS) resolution is needed. A particularly glaring lack is in the area of track fusion in a multi-target environment. Until such algorithms are developed and evaluated, the degree of effectiveness of BSTS will be open to question.

3. Discrimination

Work is needed in the area of boost-phase discrimination (missile typing) from an algorithmic point of view. Considerable effort has been devoted to the midcourse-terminal discrimination area, but we have seen only one approach to the problem of estimating the number of RVs and decoys. Most of the work has concentrated on the observable fusion and classification-determination problems. More effort is needed to include environmental effects; all the algorithms assume perfect knowledge of the statistics of random variables.

4. Common Reference Frame

There is no ongoing algorithmic work to establish a common time base and geographic grid for a multi-platform system. It is assumed that coordinated handover from one moving platform to another is errorless. At a minimum, a study of the impact of such errors on performance is required. For example, what is the impact of relative sensor position errors on track fusion in a dense target environment? This should be followed by development of methods of time and common grid dissemination in an environment of many platforms and a derivation of the expected errors of these methods.

5. Threat Assessment

Threat asessment is a critical area that has not received the attention it deserves. In many of the analyses the threat is generally simple and canonical, with no surprises for the defense. We suggest that more technical Red-Team input be provided to the contractors so that they have some realistic threat structures to analyze. A systematic breakdown of threats into categories that allow their pairing with meaningful defense postures is needed. This should be followed by development of methods for clarifying emerging threats into these categories, based on the observables and other information (e.g., intelligence).

6. Weapon-to-Target Assignment

The weapon-to-target function exhibits more activity than any of the others described here, and rightly so, since even the crudest assessment of an architecture must include some method of weapon-to-target assignment.

For the early phase of deployment, optimal weapon-to-target assignment algorithms will not be a critical battle-management function. In an interceptor-poor architecture, almost any reasonably efficient method of pairing interceptors and targets will suffice. It does not matter if 60 percent or 70 percent of the RVs leak, because the mission of the early phase is only to introduce some degree of uncertainty to the Soviet attack planner. Thus, in the near term, an extremely sophisticated algorithm is not essential to the success of the mission.

However, in the later phases, the mission becomes one of a more robust defense, with a high level of confidence in the protection of certain target classes. Here, sophisticated algorithms are necessary. If the defense can destroy several hundred more RVs simply by using a better algorithm, it may be well worth the price of extra computing capability, given the cost in Space-Based Interceptors (SBIs) needed to achieve the same increase in defense.

Efforts to develop novel assignment algorithms are under way. The earliest efforts focused on the boost and post-boost phases of the battle, but with the emergence of the Army's experimental version, a need for good midcourse and terminal algorithms arose and this need is being filled. One area which appears to lag behind is the development of strategies and algorithms for the SDS to defend against an ASAT attack.

7. Mode Selection

Any systematic approach to mode selection must go hand-in-hand with threat assessment and categorization (see above), the result of which is then used to choose a strategy from a library of respones. Making this process dynamic will lead to a strategy which is adaptive to a changing threat.

8. Engagement Selection

No significant work was identified in this area, primarily due to the fact that such a capability would be needed primarily in a dynamic, time-varying engagement scenario, and such a capability has not yet been sufficiently developed or analyzed.

9. Sensor Resource Management

The work relating to resource management is limited to methods of forming battle groups into which the sensor platforms are subdivided. In most instances these sensor groups are static. What is lacking is a time-line analysis of the sensor-resource requirements and their efficient use. Many of the planned battle-management testbeds would be extremely helpful in executing sensitivity studies of tracking performance, discrimination performance, and defense robustness to ASATs. Since the outputs of this function are largely dependent on the threat, detailed Red Team scenarios for simultaneous launches, staggered launches, and combined ASAT/ICBM launches are necessary for full evaluation.

E. CONCLUSIONS

The level of activity in developing algorithms for basic BM/C^3 functions is uneven. For a detailed breakdown, see Fig. S-1. The following tabulation presents an estimate of the adequacy of the effort by functional area.

Function	Level of Effort
Resource basing	Sufficient
Track initiation and maintenance	Sufficient
Discrimination	Sufficient
Common reference frame	Nonexistent
Threat assessment	Too light
Weapon-to-target assignment	Sufficient
Mode selection	Too light
Engagement assessment	Too light
Sensor resource management	Too light

A detailed evaluation of the algorithms being developed in each of the nine functional areas is given in Section D. Further effort is needed in the following areas:

- Algorithm for weapon-to-target assignment in an ASAT attack, or a combined attack
- The effect of various clustering strategies of weapon platforms into battle groups
- Algorithms and strategies for an attack in waves (non-spike)
- Interfaces and handover between layers, ground- and space-based assets in particular (e.g., track files, discrimination, and dynamic data)
- Algorithms which operate near optimum not only under massive but also smaller attacks
- A convincing demonstration of track initiation, maintenance, and track fusion. This is a capability vital to any deployment plan. In addition, the complexities of midcourse tracking (e.g., bulk filtering, tracking of the large number of objects) are daunting and largely unexplored.

In general, the algorithms lack flexibility, universality, and portability, leading to a number of problems.

-- Algorithms are often developed in the context of a specific architecture or overall software configuration/testbed, hence they are not testable anywhere else, and although the approach might be of interest in some other context, the software would have to be recoded.

- -- Algorithms offer partial solutions to partial problems; a developer is often partial to the aspect of the overall problem to be solved, e.g., in a weapon-to-target assignment algorithm one developer might include missile typing information derived externally, another might include the interceptor dynamics, but neither might include both.

The choice of performance goals and requirements is often arbitrary; standardization is needed in the following areas:

- Performance goals
- A baseline architecture or a set of alternative architectures which are widely disseminated (this problem may be alleviated soon)
- A clearly formulated set of alternative threat parameters
- A clearly defined set of alternatives (short- and long-range) concerning interfaces with other parts of the system (e.g., what data is made available, and where).

The dependence of the algorithms' performance on the underlying technology is critical. For example, the resolution of the sensors (or associated errors) have a major impact on algorithm performance, and sometimes even feasibility. Thus, the evaluation of algorithms must include an evaluation of the assumptions about the supporting technologies.

There is an information flow problem. Because of the accelerated nature of the program, security restrictions, and the desire on the part of corporations to safeguard their competitive edge, the timely dissemination of results useful to others is not achieved. A particularly unfortunate example of this problem is the Special Access Requirement (SAR) of BSTS, where vital information is hard to obtain. Another example is the unsatisfactory flow of information on algorithm development between the services.

F. RECOMMENDATIONS

- The following areas need further effort:
  - -- Algorithms for common time and geographic grid dissemination between platforms
  - -- Algorithms for threat assessment
  - -- Algorithms for mode selection
  - -- Algorithms for engagement assessment

- -- Algorithms for sensor-resource management
  - -- Algorithms for weapon-to-target assignment under ASAT or combined attack.

- Weapon-to-target assignment algorithms which operate at or near-optimum, regardless of the size of the attack and algorithms including effective strategies for clustering weapon platforms.

- It is recommended that each algorithm development effort be assigned to one of the following categories:
  - -- An algorithm for a specific function in the early deployment architecture with functions, assumptions, goals, and interface parameters given.
  - -- An algorithm for one of the architectures for future growth. For each of these architectures the functions, assumptions, goals, and interfaces for each algorithmic function are *a priori* defined.
  - -- A free-roaming investigation with innovation and creativity in mind.

- It is recommended that as part of the support rendered into the architecture development process by the algorithm development activity, for each candidate architecture, and for each algorithmic function, a clearly defined set of assumptions, factors to be included, requirements, threat parameters, inputs, and outputs be defined.

- Overall full-time technical coordination of the algorithm development effort is needed. Its main functions would be:
  - -- To monitor the various algorithm development activities, detect deficiencies in the overall effort, and recommend remedies
  - -- To insure that the architectural concepts will have sufficient algorithmic support at their disposal when implemented
  - -- To act as a broker and coordinator for the algorithm development effort by the services, and insure cohesion
  - -- To carry out the information-dissemination effort described in the next recommendation.

- Vehicles must be created for the timely and unencumbered dissemination of information, i.e., results, reports, new developments, etc. A number of measures can be highly beneficial:
  - -- Workshops that are carefully controlled by formulating topics and goals well ahead of time and limiting attendance by invitation only to technical contributors.
  - -- An ongoing library and document-dissemination effort.

- Removal of impediments created by SAR restrictions in the BSTS program on the distribution of badly needed information.
- Removal of impediments created by the reluctance of private companies to divulge information for fear of losing their competitive edge.

I. INTRODUCTION

A. PURPOSE

Given the intricacies of managing an enormously complex program such as the development of battle-management algorithms for the Strategic Defense Initiative (SDI), a program plan is a vital tool. Ideally, such a plan will provide a quick overview of and insight into the technical and administrative aspects of a given program.

The creation of a program plan for the Battle Management/Command, Control, and Communication (BM/C^3) algorithm program is divided into six major steps.

1. Define algorithm requirements.
2. Identify and analyze currently funded tasks.
3. Compare steps one and two.
4. Identify deficiencies.
5. Identify programs not currently funded by SDIO BM/C^3 or others but which may contribute to the algorithm technology for SDI BM/C^3 and may eliminate some deficiencies.
6. Provide recommendations for structuring the BM/C^3 algorithm technology program.

This Memorandum Report represents a program plan for the BM/C^3 algorithm technology program. The six steps used to create the plan have been performed in varying depth, depending on available information and resources. A program plan must be updated periodically as architectures and requirements change, as funding profiles change, and as new algorithms are developed and validated, hence this plan should be regarded as only one output of an ongoing effort.

B. BACKGROUND

The objective of the BM/C^3 portion of the overall SDI effort is to develop the technologies needed to support responsive, reliable, survivable BM/C^3 for strategic defense. These technologies are grouped into five areas:

- Battle-management algorithms
- C^3 network concepts
- Processors
- Communications
- Software engineering.

This document focuses on battle-management algorithms. Its purpose is to provide information and analysis to the SDIO to be used in formulating budget requirements for FY 1988 and beyond. Based upon analysis of BM/C^3 algorithm requirements and ongoing programs, recommendations are made for research that should be funded, including both current work and new tasks.

The Battle Management Algorithm task of the SDIO BM/C^3 program covers the following areas: analysis and research leading to the development of battle-management algorithms which are responsive to the BM/C^3 architecture requirements developed in the BM/C^3 Experimental Systems Project. Battle-management algorithms are the mathematical/logical processes and procedures used to allocate resources, manage and form track files, execute command and control actions--whether autonomous or human-in-the-loop--and generally implement the functionality of a strategic defense system in a robust manner which is responsive to change and evolving technology. Software implementation of battle-management algorithms in a loosely coupled, widely dispersed, real-time, heterogeneous multiprocessor environment is an aspect of algorithm implementation.

The nature and capabilities of these algorithms can have a substantial impact on overall defense system performance and cost. Very often, a more effective algorithm leads to a substantial reduction in equipment. A typical example would be a weapon-assignment algorithm which operates in real time to determine which missile intercepts which booster. A more effective algorithm can lead to a reduction in the number of space platforms or interceptors, without reduction in performance. Another example would be a more efficient method of computation which leads to use of a smaller processor at some node.

SDIO's BM/C^3 activity is widespread and involves many players who developed, used, or postulated algorithms to perform various functions. Some duplications in this process were inevitable, and in fact desirable, in order to arrive at the best solutions. However, the proliferation of such algorithms makes it worthwhile to undertake a review and dissemination endeavor, so that the community as a whole can benefit from the results obtained so far--and these are numerous and valuable--and to reduce needless duplication.

The purpose of this effort is to initiate a systematic review of the algorithms envisaged for SDI. The word "initiate" is used advisedly, since it is not possible to complete an exhaustive survey of even the viable algorithms; one main reason being the lack of explicit information. Algorithms are often embedded in software models; their properties, assumptions, and limitations are often implied but not clearly stated. Algorithms are also being independently developed in many other SDIO projects or programs [e.g., Space-based Kinetic Kill Vehicle (SBKKV), Directed Energy Weapons (DEW), Boost Surveillance and Tracking Sensors (BSTS), Space Surveillance and Tracking System (SSTS), etc.]. Many of the underlying assumptions are taken for granted, even by their developers. The fact that these developers are geographically far-flung adds to the difficulty. In spite of these handicaps, the task is necessary, since the potential benefits are substantial. Hence, the motivation for this IDA effort. To achieve useful results in this area without dilution of effort, other related areas--although of substantial merit--are not included. These include processor hardware implications, software system developments, and networking requirements.

A number of benefits are expected from this ongoing effort:

- To provide information and analyses to the BM/C^3 Directorate of SDIO which will enhance the effectiveness of its algorithm development effort.
- To ensure that the best algorithms will find their way into the design efforts.
- To promote technical exchange on algorithms within the SDI community, stimulate creative thinking, and enhance awareness of alternative approaches.
- To identify areas where further algorithm development is needed.

One of the difficulties in conducting an algorithm review is the development of a method of classification which allows one to see the primary function of an algorithm at a glance. Considerable effort was devoted to this aspect, the results of which are presented in the following chapter.

II. SDI BM/C^3 ALGORITHMS IN END-TO-END BATTLE CONTEXT

A. BM/C^3 ALGORITHMS IN AN END-TO-END ENGAGEMENT

In each phase of the SDI operation, both prior to and during an engagement, algorithms are used. Although their number is literally in the hundreds, the SDIO's system design activities pinpointed and formulated at this stage only those having either a critical impact on overall performance, or presenting particular problems of implementation (see Fig. 1).

Prior to the deployment of a ballistic missile defense system, the intended locations of weapon and sensor resources must be determined. Basing algorithms are used to derive resource placements which can best respond in real time to a range of postulated threats. Ground-based interceptors are positioned with respect to their intercept envelope (i.e., footprint), the assets they are responsible for defending, and the magnitude of the threat they are expected to negate. Best offense versus best defense simulations are often run to determine which locations will minimize leakage or expected damage. Ground-based sensors are placed so that sufficient tracking and assessment data can be provided to the battle manager to effectively use the interceptors.

In the real-time execution of ballistic missile defense, a multitude of battle management functions must be performed. These functions fall into three categories: organizing information, assessing information, and taking action based on the assessments. Within a given battle area, the data organization and assessment functions will be performed simultaneously throughout the battle, while specific actions will be taken intermittently as sufficient data is accumulated and evaluated.

The first organizational function is the establishment and maintenance of a common time and space reference frame. The objective is to enable the transfer of spatial information and the coordination of multiple platforms by minimizing the errors in perceived position, orientation, and local time between the various platforms. Though this

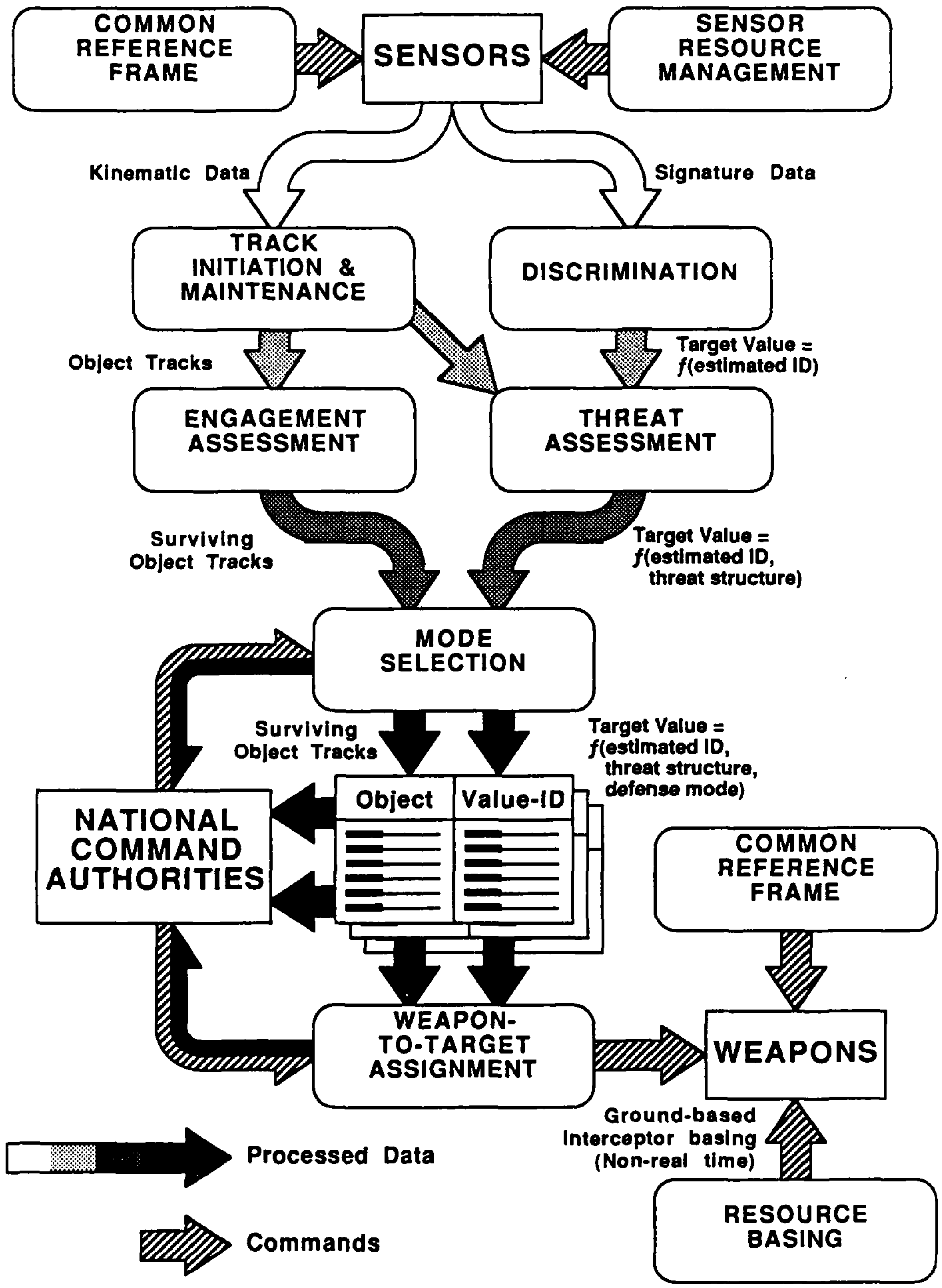


Figure 1. BM/C^3 Algorithms and Their Interaction.

will be part of the peacetime regimen of the strategic defense system, it will be most crucial during battle. Since most of the planned sensors are passive, they will provide angle-only data to the battle manager. Without range data and with the large distances between defensive platforms, errors in perceived orientation between cooperating sensor platforms would result in substantial errors in the position of an object being tracked. Positional errors in the weapon platforms would result in miscalculating which threat objects lie within the weapon envelopes. Since the platforms are in motion, time is also critical when observations from various platforms at various times are compared to create accurate tracks on threat objects, or when interceptors must be guided to the perceived threat intercept position. Thus, the common time and space reference systems are interrelated.

Once a launch is detected, the first task is to determine the number of boosters, their positions, and their kinematic profiles (i.e., form track files) so that their future positions may be estimated for possible engagement. This is done by the track initiation and maintenance algorithms. These algorithms perform three functions that are necessary to form a track: scan-to-scan correlation, state prediction (position, velocity, acceleration), and multi-sensor fusion.

Since many hundreds or thousands of boosters may be visible in a sensor's field of view, there is a problem with associating the observation of one booster in one sensor scan with the observation of the same booster in a subsequent scan. This problem is magnified by high target density, low sensor resolution, false observations (noise), missed observations, and low scanning frequency. Various scan-to-scan correlation schemes are used to properly correlate observations, with the accuracy improving with each new scan, although there is a general trade-off between correlation accuracy and computational resources (memory and speed).

Once several observations of an individual observation are correlated, the state prediction algorithms can be initiated to predict the position of the booster on the next scan (for maintaining the track) or predicting further in time to evaluate a potential engagement opportunity. Multi-sensor fusion algorithms combine track files from multiple sensors to increase track accuracy. In this operation, correct pairing of tracks from different sensors is critical.

As track initiation and maintenance algorithms organize the sensor information, the boost-phase threat assessment algorithms determine the nature of the threat. Classification of booster types enables the battle manager to assign a high priority to those booster types with the potential to carry many reentry vehicles (RVs) or decoys. For example, launch sites, plume signatures, and trajectory data may all be used to classify a booster as a high threat SS-18 (10 RVs), a lower threat SS-25 (3 RVs), or an Antisatellite (ASAT) booster. Proper identification of boosters is crucial since the selection of a defense mode or engagement strategy requires accurate knowledge of the threat.

As assessment of the threat becomes more detailed, the defense's readiness level will be adjusted by the mode-selection algorithms. Pre-battle status including Defensive Concentration (DEFCON) levels, and current military activities must be considered before a defensive posture is established. In a limited attack, such a posture might require human enabling of weapon systems. In a massive launch, the system might be placed in an "automatic" mode where the rules of engagement follow any one of several predetermined doctrines, according to the assessment. Since time is critical to the interception of missiles in their boost phase, the rapid establishment of a defensive mode tailored to the threat is vital for the strategic defense system.

Once sufficient information is available and a defensive mode has been established, the battle manager may act on this information by initiating the weapon-to-target assignment algorithms for the appropriate DEWs or SBKKVs. Engagement feasibility is based upon track files and position/ status information of the weapon platforms. For example, the interceptors on an SBKKV platform moving into range of the threat cloud may be used in many ways in an optimal weapon allocation scheme since it will be able to engage many targets. However, a platform moving out of range of the threat cloud may be instructed to fire all its interceptors immediately, since they will be useless once out of range. Boosters identified as carrying large numbers of RVs or decoys (SS-18s) might be preferentially engaged to minimize the tracking and intercept burden in subsequent layers. ASATs further complicate the engagement scenario since the defense must decide the proper fraction of resources to devote to self-defense to maximize the number of boosters killed during the boost phase, or the probable number of RVs killed in successive phases of the defense.

If there is time for shoot-look-shoot strategies (e.g., in midcourse), the organize/assess/act cycle comes full circle, and the engagement assessment algorithms are used to determine whether an intercept actually occurred for each engagement assigned. If an intercept has not occurred, the battle manager may arrange for additional engagement attempts. If the engagement is judged a success, the battle manager must deal with any additional tracks created by fragments resulting from the intercept.

This cycle repeats continuously through the post-boost, midcourse, and terminal phases of the battle, although there are some additional functions which must be performed, as well as complications to those already listed. Beginning in the post-boost phase, the number of objects to track will increase by several orders of magnitude. Since the RVs and decoys are deployed onto ballistic paths, there is no easily visible booster plume for the sensors to track. As new objects are deployed, new tracks must be initiated for each, and it will take some time before these tracks are accurate enough to plan an engagement. For these and other reasons, the post-boost and midcourse track initiation and maintenance algorithms will have to be quite different from those used in boost phase, and much more complex.

As RVs and decoys are deployed and fan out from their launch points, the space that must be monitored increases substantially. Therefore, sensor resource management algorithms must be used to form the SSTS satellites into groups for multi-sensor fusion and surveillance volume responsibility. Due to the large number of objects that will be required to be under surveillance, sensor resources must be carefully allocated.

Before any engagements can be planned between the post-boost and terminal phases, discrimination algorithms must analyze the sensor data on each track to determine the probability that the tracked object is an RV or a decoy. There are many discrimination schemes, each involving three critical issues. First is the question of how multi-phenomenology data (three-color infrared, ultraviolet, laser radar imaging, and neutral particle beam returns) should be combined to maximize the probability of distinguishing between RVs and decoys. Once this is done, the algorithm must use the combined data to determine how many classes of RVs and decoys exist in the threat cloud and how many of each type there are. Finally--and most critical to the battle manager--the algorithm must assign priorities to individual targets for engagement. Depending on the types of information that other battle-management algorithms (e.g., weapon assignment) require, the algorithm could provide the probabilities of each object being an RV, ranking targets from most to least lethal, or tag each target as RV or decoy. If an active discrimination technique or if staring sensors are used, scheduling algorithms must also be used to determine the optimal order in which to scan the objects in a threat cloud. Another point useful to the battle manager is how discrimination ability changes with time. If the ability increases, the battle manager may choose to give up some engagement opportunities to allow time for more accurate RV classifications. To handle the large number of objects in space (e.g., debris), the above tasks must be combined with bulk-filtering, which reduces the number of objects to be considered.

In the late-midcourse to terminal layers of the defense, aim-point threat assessment is necessary to the planning of an efficient engagement. As impact-point prediction data becomes available, a preferential defense may be exercised to defend high-value targets. Some items to consider here are: hardness of high-value sites, probability than terminal defenses at each site will be overwhelmed by probable RVs in track, and any changes in site value due to RV impacts.

Engagement planning in the late-midcourse and terminal layers of the defense is complicated by two problems. First, since these are the last layers, any leakage must be "mopped up" on a site-by-site basis; 100 percent attrition of RVs is necessary for a successful local defense. Second, since positions of the defense interceptors are fixed and the threat is coming at them, a given site might have a multiple shoot-look-shoot capability, but due to kinematics the area that site can defend (its footprint) shrinks with time. The terminal defense process can be greatly improved by coordinating space-based defenses, including weapon assignment and the handover of midcourse discrimination information. Such coordination, however, greatly complicates the battle manager's task of conducting an efficient and effective engagement.

The above descriptions apply to a wide variety of battle-management algorithms. Separate algorithms will most likely be necessary for each sensor and weapon system type for each phase of the battle, with several versions of adaptive features to respond to changes in defense and threat status as the battle progresses.

B. SUMMARY OF ALGORITHMS BY FUNCTION

A breakdown of algorithms by function must be attempted to achieve a semblance of order even though the results are doomed to be unsatisfactory in more than one respect. First, the mere question of what is an algorithm leads to highly subjective answers. In a system as complex as SDI, there are hundreds of functions employing some logical or mathematical procedure. A survey such as this must be limited to those which have a major impact on system hardware requirements and on top-level performance. To take a single example, even the establishment of a single link between two space platforms requires a large number of logical steps, such as antenna pointing, signal acquisition, synchronization, ranging, etc. Even after these detailed, low-level procedures and operations are accomplished, interrelated subdivisions of the major functions must be performed. For example, weapon-target assignment implies a chain of command, definition of jurisdictional areas, various data-base manipulation, common time and space references, discrimination functions, and guidance, to name a few. Obviously, there is more than one way to configure these operations into groups, each defining a type of algorithm.

The following functional breakdown, used throughout this report, is based primarily on the activities of various organizations that work on algorithm development; a gradual, quasi-pragmatic subdivision emerged within this community and on our perception of a logical breakdown into functions. The following eight functional areas are identified.

- ***Resource basing,*** which is concerned with the deployment of ground-based interceptors for point and area defense.
- ***Track initiation and maintenance,*** which pertains to establishing, and by implication, predicting the tracks of threat objects from sensor observations.
- ***Discrimination*** pertains to classifying the threat objects (usually after ejection from a post-boost vehicle) into reentry vehicles (RVs) and decoys of different types.
- ***Common reference frame*** pertains to the establishment of a common spatial and temporal coordinate system between platforms to synchronize operations and transfer positional data.
- ***Threat assessment*** involves the estimation of relevant parameters of a threat as it develops.
- ***Weapon-to-target assignment*** involves the assignment of weapons to threat objects.
- ***Mode selection*** involves selection of the overall strategy most appropriate to the threat.
- ***Engagement assessment*** involves evaluation of the results of engagements as they develop.
- ***Sensor resource management*** entails the allocation of the overall task between the sensors to achieve efficient operation.

These algorithms are described in more detail on the following pages.

III. DESCRIPTION AND REVIEW OF ALGORITHMS

In the previous section, nine classes of algorithms were identified as being of particular relevance to SDI. In this section each of these is discussed in detail. A brief description will be given of the application, function, and role of the algorithm within the engagement scenario. The key factors, assumptions, figures of merit, and methods of computation are discussed in separate subsections. Each of the four subcategories merits a brief explanation.

In developing an algorithm, one must take into account the real-life situation prevailing during its use. Factors such as time delays, propagation phenomena, natural effects, and features and capabilities of the threat objects may have an impact on the performance of the algorithm approach used. Relevant factors differ from algorithm to algorithm, but a checklist is a handy tool when trying to decide if a new algorithm considered all facets of the real-life situation.

Many of the elements relevant to the engagement are fraught with uncertainty; this applies not only to natural phenomena which are not completely understood but also to the capabilities and strategies of the opponent. The assumptions used for these elements must be clearly stated. If this is not done there is the danger that, in comparing two algorithms, results may be misleading. For example, the seemingly less efficient of the two might in fact be the most efficient although it appears to yield poorer results, because it is based on more stringent assumptions.

Figures of merit are attempts at numerical quantification of the power of the algorithm. The implication here is that, all other things being equal, the algorithm with the higher figure of merit yields superior performance. This does not mean that the algorithm is preferable since the superior performance might have been purchased at some price, such as higher computational load. The method of computation is a convenient means of categorization, since only a limited set of approaches is available for many classes of algorithms.

After a discussion of the elements in each of the four categories listed above, for each class of algorithms the discussion concludes with a description and analysis of the specific algorithms surveyed.

A. RESOURCE BASING

1. Description

Algorithms for interceptor basing predate SDI. They were developed for air defense applications. In the SDI context, relevant components of the problem are the following:

- A set of targets of different values to be defended.
- A given number of attacking RVs, each aimed at a specific target.
- A given number of interceptors. There could be more than one type of interceptor, and a number of interceptors of each type. For each type of interceptor a set of characteristics and capabilities are defined, such as kill probability or footprint of the defended area vs time.
- In addition to the attacking warheads, there could be decoys of different types. Associated with each type of decoy there is a discriminator parameter, K, which defines how well a decoy can be distinguished from an RV. K could be time-varying.
- A system which provides impact-point prediction for each incoming threat object. Accuracy of the prediction could increase with time.
- Implied in the shrinking footprint and different capabilities of the different types of interceptors is a layered defense, with a different type of interceptor used for each layer. One example of such a situation is a three-layered defense consisting of Exoatmospheric Reentry Vehicle Interceptor Subsystem (ERIS), High Endoatmospheric Defense Interceptor (HEDI), and Low Endo-atmospheric Defense Interceptor (LEDI). Depending on the interceptor capabilities in some layers, a shoot-look-shoot or even a three-shot capability might be available.

The problem is reduced to a game between two players. One would seek to minimize and the other to maximize the total value surviving. Both would seek the optimum strategy to achieve this goal, and the strategy might depend on additional factors and assumptions not listed below. For example, the amount of information available to the two players is important. The attacker might have only partial information on the number of defending interceptors of different types, their capabilities (footprints), the value of defended assets, and the discriminator parameters. The defender, on the other hand, might or might not know the exact number of RVs and decoys, the number of threat objects held in reserve, and the exact value of the discriminator parameters.

Investigators have looked at different subsets of the problem; for example, decoys might not be present, or only one type of interceptor might be available. Alternatively, additional subtleties could be included, such as camouflaging empty silos to make it appear that they contain ICBMs worth attacking.

Basing of interceptors is decided prior to the engagement; by contrast, earmarking of interceptors to specific RVs is done in real time and is based on the attack distribution of the defended assets. This second part of the strategy is more properly considered weapon-to-target assignment, and has much in common with algorithms used in conjunction with space-based kinetic energy and laser weapons discussed in a subsequent section. In the case of ground-based defense, weapon-to-target assignment and basing are interrelated and many algorithms treat them as a single problem.

Analyses of ground-based terminal defense problems involving abstract mathematical constructs, e.g., minimax problems requiring linear programming techniques, abound. They are characterized mostly by an air of unreality, and disregard of important real-life factors. Some important aspects of the problem requiring further study are the following:

- The geographic dispersion of the targets to be defended, and inclusion of their value or category (silos, cities, etc.)
- The dynamic characteristic of interceptors and their shrinking footprint with decreasing commit altitude
- The impact of time-varying discrimination quality for various pen-aids
- The random nature of the threat dispersion because of attrition in previous, e.g., boost-phase, encounters.

The following section lists critical features that are relevant when evaluating a basing algorithm.

2. Critical Features

a. Factors

Basing algorithms may be characterized by the factors and assumptions entering into the formulation of the problem. Many approaches contain drastic simplifications because some formulations have no known solution or the solutions require an inordinate amount of computer time. Identifying the factors and assumptions on which the proposed solution is based is important in evaluating any algorithm.

- ***Attack size.*** If the attack is in waves, the size of each wave as well as the size of the total assets available to the attacker is relevant. If decoys are assumed to be present, the exact mixture affects the problem.
- ***Defense size.*** If more than one type of interceptor is assumed, the number of each type is important.
- ***Interceptor performance parameters.*** Interceptor performance parameters are relevant to the extent that they permit the computation of the footprint of the area that the interceptor is capable of defending. The footprint is a function of the time available for intercept.
- ***Threat object reentry trajectory.*** The threat object reentry charcteristics also affect the footprint. In this context, the reentry angle and azimuth as well as aerodynamic characteristics are significant.
- ***Defended asset parameters.*** The three most significant defended asset parameters are the defended target value; its hardness, which determines the probability of its destruction; and deceptive basing. The last refers to the fact that a defended target might be made to look more or less valuable than it actually is. For example, an opponent might be denied knowledge about whether a silo is empty.
- ***Impact point prediction capability.*** The accuracy of impact point prediction capability is particularly important in the presence of a point defense capability.
- ***Discrimination capability.*** Discrimination capability may be assumed to be static or improving with time. The latter has a major impact on multilayer defense strategies. The numerical characterization of performance, e.g., K-factor, is important.

- ***Interceptor kill probability.*** Interceptor kill probability can be included to various degrees of sophistication, with a value of unity at one extreme to kill probability dependent not only on the specific type of interceptor but also on the geometry of the intercept approach. A compromise often used is a fixed value.
- ***Prior attrition of threat objects.*** If the ground-based defense is the last layer of a multilayer engagement, for example, in a system with a boost-phase defense, the RVs aimed at ground-based assets undergo prior attrition. This attrition involves random elements which greatly complicate the strategy of the attacker as well as that of the ground-based defender. An optimum algorithm must take into account the probable distribution of the reentering RVs.
- ***Attack timing.*** Here the most important question is whether the attack is simultaneous or in waves. The algorithms for the two cases could be significantly different.
- ***Knowledge available to the two opponents.*** Any algorithm which does not state assumptions concerning knowledge of the two players, or one based on unrealistic assumptions, is seriously deficient. The list of such elements includes nearly all elements of the scenario. To name a few, the opponents may or may not know the exact size of the battle assets (number of RVs and interceptors); the two opponents might attribute different values to different defended assets; the values of kill probabilities used by the two sides might be different, and both might be incorrect; and discriminator performance might be seriously misjudged by either or both.

b. Assumptions

Many of the basing algorithms incorporate assumptions that have a significant impact not only on the results, but also on the method used to solve the problem. These assumptions are introduced either because there is not enough information available about the situation that will prevail in reality, or because an exact solution is not possible or not practical. The lack of sufficient information can result if the phenomenology is not properly understood or if the actual conditions which will prevail depend on future events and decisions. The following is a limited list of assumptions which were occasionally adopted by the developers of algorithms; since each developer uses his own judgment about the assumptions and simplifications he introduces, these are highly judgmental and any such list is, by its very nature, incomplete.

- ***There are no decoys.*** This assumption is tantamount to postulating a perfect discrimination system with no delays.

- ***Perfect kill probability.*** This assumption changes substantially not only the outcome of the engagement, but also the structure and methodology of the basing algorithm.
- ***Perfect impact-point prediction.*** While the previous two assumptions are arguably too optimistic, impact point prediction might conceivably be accurate enough that any further improvement would not significantly affect performance.
- ***No shrinking footprint.*** This assumption does not affect the results if there is no shoot-look-shoot capability. There are many simplifying assumptions which reduce the time-varying dynamic problem to a static one. For example, one might postulate a system with area defense for the first shot and point defense for the second.
- ***Perfect knowledge of the number of threat objects and their distribution between RVs and decoys of different types.*** An algorithm taking count of the uncertainty in these parameters could be substantially different from one that does not.
- ***Discriminator performance.*** Assumptions concerning discriminator performance can take many forms; for example, one may assume there is no discrimination prior to the first shot and perfect discrimination for the second shot. This would be the case if light balloons are stripped away upon reentry. See also the statement above under "There are no decoys."
- ***Interceptor Vulnerability.*** The incoming RVs may destroy not only the defended assets, but also the interceptors. This can affect the defended value surviving either if one is dealing with a wave attack or if there is a cross-defense capability (e.g., area defense). In these scenarios it is important to include the impact of RVs on interceptors.

c. Figures of Merit

The ultimate objective of any basing algorithm is to maximize the value surviving. Hence, a figure of merit, obtained with all other things being equal, which measures this or some related parameter is most frequently used. Of course, other parts of the defense configuration, such as the weapon-to-target assignment algorithm, also have an impact, therefore these figures of merit should be used only for comparative purposes.

- ***Percent RV leakage.***
- ***Number of targets surviving.***
- ***Defended value or percent defended value surviving.***

- ***Mix of assets surviving.*** This measure is based on the thought that an effective defense strategy must aim at the survival of some of each type of asset, e.g., retaliatory capacity of command structure, hence a linear combination of surviving assets is not an optimum measure of effectiveness.
- ***Cost trade-off.*** This is some measure which trades off the expenditure in interceptors (e.g., cost or number) against the number of attacking RVs for some level of survival.

d. Computational Methods

It is beyond the scope of this effort to describe the mathematical techniques that have been used. Depending on the formulation of the problem, some are more suitable than others. They usually fall into the following three categories:

- ***Linear programming.*** This is one of the most frequently used approaches, not only for these algorithms but also for weapon-to-target assignment.
- ***Lagrangian multipliers.*** The problem in this case is complicated by the fact that we are dealing with solutions in an integer space.
- ***Branch and bound method.*** This approach consists of a tree search in which unlikely solutions are rejected through comparison with theoretical bounds.

From the defender's point of view, a solution consists of determining the basing of interceptors at different defended targets; if an interceptor can defend more than one target, the strategy for allocating interceptors in real time to the defense of specific targets is of interest. If shoot-look-shoot capability is available, the optimum allocation of interceptors to different shots may be suitable. The attacker has the similar task of determining the optimum allocation of RVs to targets. The solution usually is a minimax solution in that each side optimizes its strategy to minimize the effects of the optimum strategy available to the opponent.

3. Review of Resource-Basing Algorithms

Because of the challenging mathematical problems involved, there is a great deal of literature on the subject of resouce basing, much of which antedates SDI and was a product of early ABM studies from the 1960s. Different algorithms were developed to cover different subcases such as:

- Attacks in waves
- Shoot-look-shoot allocation

- Salvo strategies
- Shrinking interceptor footprints
- Impact-point prediction.

a. MVADEM[1]

The MVADEM program models the following elements of a terminal defense scenario:

- A defense inventory is two-layered, consisting of a fixed number of overlay and underlay interceptors. The attacker assumes two inventories which could be the same or different from the true inventories.
- Each of the two layers of interceptors has a single-shot (or salvo) opportunity, i.e., each RV can be intercepted first by the overlay and subsequently by the underlay interceptor. For both layers, the interceptors are dedicated to targets (point defense).
- Once the basing is committed by the defender, it cannot respond to an attack different from the one it has assumed. It must defend with the predetermined strategy. This is a major limitation. In addition to single-shot capability, salvos can be included. For a salvo of N interceptors, a miss probability equal to the miss probability of a *single interceptor* at power N is inserted. The actual number of interceptors equals N times the number of interceptors used as an input parameter for the run.
- The defended assets consist of discrete groups and some number of targets in each group. Each group is distinguished by a value for the targets in that group. Targets in different groups are distinguished by different parameters, such as value and hardness (expressed as the probability that the RV will kill the target). The values attributed to the group may be different for attacker and defender, hence the defense and attack strategies will depend on the values postulated by each side. Neither side knows the other's value assumptions. In addition, the defender may camouflage some target groups to make them appear to be groups of a different value.
- Each side assumes certain values for relevant parameters, such as K factors, kill probabilities, etc. These values may or may not be correct, but they are used to develop the respective strategies.

---

1 For details of this model the reader is referred to the "MVADEM User's Guide and Reference Manual" by John C. Key, SPARTA, Inc. Prepared for U.S. Army Ballistic Missile Defense Command, Huntsville, Alabama, April 1984.

- Neither side knows the actual deployment of the other, that is, for any specific target, how many attackers or defenders are allocated?
- Each side assumes some attrition caused by previous layers, and plans its strategy accordingly.

The scenario described above can be put into a mathematical framework, leading to an allocation algorithm. The problem may be stated from the attacker's and defender's points of view as follows:

- ***Problem Statement From Attacker's Point of View:***

  Given my best estimates of

  - Total exo and endo interceptor inventory (overlay and underlay)
  - Target structure (number of groups, number of targets, values)
  - Relevant system parameters (K factors, PKs, etc.)
  - Depletion on previous layers and leakage (including decoys)

  How shall I target my inventory to minimize the expected value surviving under the most effective defense deployment?

- ***Problem Statement From Defender's Point of View:***

  Given my best estimates of the same parameters as above, how shall I deploy my exo and endo inventory to maximize the expected value surviving under the most effective attack deployment?

  The problem may be expressed in mathematical terms:

$$VS = \underset{a_1}{\text{Min}} \; \underset{d_{JK}}{\text{Max}} \sum_{g} \sum_{K} \sum_{J} d_{JK}^{g} \sum_{I} a_1^{g} \, PS(I,J,K)^{g} \cdot VAL \cdot SD \ ,$$

where

VS	is the surviving value
$d_{JK}{}^{g}$	is the fraction of targets in group "g" assigned J overlay interceptors and K underlay interceptors
$a_I{}^{g}$	is the fraction of targets in group "g" assigned I reentry vehicles, and

$PS(I,J,K)^g$	is the probability of survival of a target in group "g" when it has been assigned I RVs, J overlays, and K underlays
VAL_g	= target value in group "g".
SD_g	= number of targets in group "g".

The terms $PS(I,J,K)^g$ in the probability-of-survival matrix are computed from the various system parameters such as reliability, K factor, probability of kill, etc. Given this matrix, as well as VAL_g and SD_g, the solution to the problem is the defense and offense deployment given by the first two sets of variables defined above. In general, there is a unique solution represented by a set a_I^g for which the set d_{JK}^g is the value maximizing VS and, conversely, for the set d_{JK}^g, a_I^g is the set of values maximizing VS. In other words, d_{JK}^g and a_{Ig} represent the minimax solution.

Three cases may be distinguished:

- Both the attacker and defender have the same value of parameters, in which case VS represents the minimax solution.
- Either the attacker or the defender operates with a subset of false parameters. In the first case, the value of survivors will be larger, in the second, smaller than the minimax value expected with the same value of parameters.
- Both parties have the same false subset of parameters, in which case the value of survivors may be smaller or larger than the minimax surviving value.

The program evaluates the value surviving under any of the above assumptions.

b. R.M. Soland (George Washington University) Algorithms for Area and Point Defense of Targets of Different Values

These algorithms are included because they are representative of a substantial body of work devoted to ground-based defense problems. The work is based on linear programming and other related techniques. The four algorithms to be summarized were presented in a draft paper dated June 1985.

Given:

- The number of defended targets, T
- The number of point-defense interceptors at each defended target
- The total number of area defense interceptors
- The total number of attacking RVs

the problem is to find the distribution of the RVs inflicting the maximum damage. There is a shoot-look-shoot opportunity after intercept by area interceptors. The defended targets are of different values, and the percentage of expected damage is an input function of the defense and offense levels at that target. The interceptor kill probability is less than unity. There is no impact-point prediction.

Although the problem is here formulated from the point of view of the offense, it is relevant to basing problems, since the solution yields the maximum damage for any basing. The problem is translated into a set of linear equations with constraints, and is solved through linear programming methods.

The second algorithm is almost identical to the above, except that there is impact-point prediction prior to engagement by the area defense. In the third case, the defense consists of point defenders alone, otherwise, the conditions are the same. The fourth and last algorithm considers L non-overlapping defense areas, each with its own area defense. This situation would arise with interceptors having limited footprints.

As in the first algorithm, the problem is set up as a set of linear equations with constraints. No codes or flow charts are given, and there is no indication that any of these algorithms was ever implemented.

c. Ground-Based, Two-Layer Defense Algorithm by M.V. Finn

The two algorithms developed by Finn treat the following problem:

- The offensive forces consist of A, perfect, identical attackers.
- The defense consists of an area defense layer followed by a point defense layer, both with perfect interceptors.
- There is impact-point prediction.
- The point-defense interceptors are preallocated to defended targets of different values v(i) i = 1,2,....

The first strategy developed by Finn is the shoot-to-kill strategy. The point defense consists of v(i) interceptors at defended target "i"; the attacker selects a set of targets to attack and each is attacked with v(i)+B+1 RVs where B represents the number of area interceptors. This means, in effect, that each attacked target will be destroyed. The attacker picks his targets in descending order of value until he runs out of RVs.

The shoot-to-kill strategy is optimum for the offense if either the attack level is small and/or the number of interceptors is small. When this is not the case, the defense dilution strategy is called for.

Since the defender has impact-point prediction he will have to allocate area defenders equal to the excess of RVs over point defenders to save a target. He will select those targets to defend for which the value saved per area defending interceptor is maximum. The attacker uses a proportional strategy that consists of allocating a number of RVs proportional to the target value. This description ignores the restrictions due to integral solutions which are treated in Finn's paper.[2]

B. TRACK INITIATION AND MAINTENANCE

1. Description

The track initiation and maintenance algorithms provide the battle manager with three-dimensional position, velocity, and other kinematic data on each object within the surveillance envelopes of the ground- and space-based sensors. This information is used by the battle manager for two distinct tasks. First, the number of objects, their launch points, aim points, and the characteristics of their trajectories are used by the threat-assessment algorithms to classify specific objects and to draw overall conclusions as to the structure and significance of each portion of the attack. Second, the predicted positions of each object are used to determine potential engagement opportunities from the various weapon platforms and to compute intercept trajectories.

A wide number of sensor types is available for each battle phase and phase of deployment for the strategic defense system. Although these systems may be active or passive and their performances may differ greatly, the fundamental goals of their algorithms are the same. The algorithms must perform a series of functions associated with the problem of tracking multiple targets:

- Scan-to-scan correlation
- State estimation and prediction
- Multi-sensor fusion.

---

2 Michael V. Finn, *The Value of Area Defense Impact Point Prediction in a Two-Layer Defense with Perfect Attackers and Defenders*, IDA Paper P-1902, September 1986.

Since most of the planned sensor systems are scanning sensors, threat observations will come in frames or scans separated by some small scan time. If two successive scans show the positions of 1000 objects, a total of 1000^2 or 1 million possible associations are possible of the objects in the first scene with the objects in the second scene. This is the scan-to-scan correlation problem. Some of these associations are unrealistic and can be eliminated. For example, the maneuvers between frames that a target would have to execute for the association to be correct do not agree with the modeled capabilities of the target. Even so, scans producing a dense cluster of targets in the focal plane of the sensor will produce a number of possible associations for each target. With each new scan, some false associations are eliminated since they are rendered less and less likely by the new data. There is a trade-off between reducing computation, but accidentally eliminating true tracks or by retaining too many false tracks, thereby necessitating an excessive computational load. Algorithms aiming at a compromise between these conflicting goals are being developed for the SDIO. They present a heavy computational load, especially in midcourse, where the number of objects being tracked is very large. Trade-offs between sensor accuracy and computational complexity also must be addressed. For example, in the limit, perfect noise-free sensor data will permit relatively simple tracking algorithms, but noisy sensor data with a number of unobserved targets on different scans will require complex algorithms which may yield poor results.

The state estimation and prediction process is common in the single-target and multiple-target tracking fields. After a series of correlated observations, the position of the target on the next scan can be predicted by a variety of means, such as Kalman filtering or template matching. These estimation techniques are such that the error in predicted position decreases as additional correlated scans are accumulated for each target, and is therefore useful in the target correlation problem discussed above. An important point is that the outputs of the state estimation process are generally position and velocity in three dimensions, even though many of the planned sensors for strategic defense are passive and will provide angle-only data.

After tracks are established separately on each platform, fusion of the tracks generated at different platforms yields further improvement. Data from widely separated sensors would aid in resolving the range ambiguities of observed targets, but this requires very-high-data-rate communications and coordination between sensor platforms. Because of these complications, the community has generally taken the approach of forming three-dimensional tracks on each sensor platform by making several observations of a target cluster before attempting track initiation. Since the estimated track file data is considerably less than the sensor focal plane data necessary to generate the track, the track files from multiple-sensor platforms can be transmitted to a central point where the multi-sensor fusion algorithms combine corresponding tracks from each contributing sensor. The resulting fused tracks are more accurate than those formed by a single sensor, and are used by the battle manager for threat assessment and engagement planning.

It is generally acknowledged that the tracking algorithms are the most time- and computationally-intensive of all the strategic defense battle-management algorithms. Any approach to this problem will represent a trade-off between accuracy, resources required, and timeliness. Very accurate tracks on boosters may be available after observing them through most of their boost phase, but by then it is too late to use the information to assign a weapon for engagement since the plume may not be visible or most of the booster's payload will have been deployed by the time an engagement can take place. Accurate correlation of a very dense threat may be achievable by a given algorithm within the necessary time constraints. However, this may require a computer the size, power requirements, or cost of which may make its deployment prohibitive. Obviously, development and evaluation of tracking algorithms for SDI will require thorough integration with other battle-management functions, and sensor and sensor-platform design.

2. Critical Features

a. Factors

- ***Frequency of false observations.*** Some observed "targets" in the focal plane of a tracking sensor may represent noise, clutter, or jamming, rather than an actual target. This parameter is often referred to as "Probability of False Alarm" (PFA) or "False Alarm Rate" (FAR). False observations can mislead the correlation and state estimation algorithms, requiring that the true track be recovered or reinitiated once corrupted.
- ***Frequency of missed observations.*** If a complete data set is not available on the threat cloud at each scan, the tracking algorithm must make provisions for maintaining a track based on previous information.
- ***Background and sensor characteristics.*** This relates to the first two factors. If the background and sensor characteristics are known, the tracking algorithm should make provisions for them, adjust the track-forming process when necessary, and be aware of the performance costs of these adjustments.

For example, while a sensor is scanning a threat cloud it may pass in front of a bright cluster of stars. If the algorithm knows this, it can "subtract" the cluster from the focal plane so it is not taken to be a target. Otherwise, the detection threshold in the focal plane may be raised so that stars are not admitted as targets, even though this will increase the chances of missing an observation of a target, as described in Factor 2. Including these characteristics allows an optimum dynamic trade-off between false observations and misses.

- ***Sensor coordination errors.*** During multi-sensor fusion of tracks, track-state errors will accumulate if misalignments between contributing sensor platforms or individual platforms and a common reference frame are not accounted for.
- ***Track splitting.*** As RVs and decoys are deployed from the bus during the post-boost phase, the bus track will be splitting off into new tracks, one for each object deployed. Provisions should be made for these phenomena, and the battle manager should provide the tracker with estimates of when deployment is most likely to occur based on missile typing information (see Threat Assessment, Section E).

b. Assumptions

- ***Perfect observations.*** All observations in the focal plane of the sensor result from targets, and all targets within the field of view of the sensor are observed on each scan.
- ***Perfect sensor alignment.*** No alignment errors between sensors are accounted for in the multi-sensor fusion algorithms.
- ***Perfect kinematic model.*** The target kinematic model (maneuvering characteristics) used in the state-estimation and prediction algorithms is known. There is no need to adjust and update the model as more information becomes available.
- ***Perfect scan-to-scan correlation.*** No successive observations of a target are incorrectly associated with observations of another target or noise.
- ***Number of targets is known.*** If the number of targets is known with certainty, the problems of false or missed observations is reduced.
- ***Threat structure is known.*** In the boost phase, if the launch is known to be staggered, new tracks will have to be initiated over the launch interval. If a simultaneous launch is assumed, the problem of initiating new tracks is reduced.

c. Figures of merit

- ***State covariance matrix.*** Minimize the error in the predicted state of the target on the next observation by minimizing the state covariance matrix (Kalman filtering).
- ***Number of false tracks formed.*** Minimize the number of false tracks formed in the track initiation process.
- ***Number of false tracks maintained.*** Minimize the number of false tracks maintained due to miscorrelations between scans.
- ***Track initiation time.*** Minimize the time required to initialize a track.

d. Computational methods

- ***Kalman filtering.*** Predicts future states, once scan-to-scan correlation is completed, by finding the state that minimizes the state error covariance matrix.
- ***Munkres or "Hungarian" algorithm.*** Performs the scan-to-scan correlation function by assigning observations to specific tracks. The algorithm is optimal, as measured by the cost function used in the assignment process.
- ***Tree trimming.*** This tracking algorithm maintains many possible tracks, including miscorrelations, then trims off tracks that do not meet some set of specifications after a given number of scans.
- ***Probability Data Association Filter (PDAF).*** Performs track correlation by producing a new track based on a weighted average of individual observations.
- ***Track templates.*** Performs state prediction and track correlation by placing nominal trajectory templates over "maps" of all the observed target states.

3. Review of Algorithm Development Activity

A variety of algorithms are being developed for the multi-target tracking function of SDI. Some contractors have chosen to examine the problem a step at a time, slowly increasing the fidelity and therefore complexity of the algorithm (Mitre, ATA) while others have chosen to solve the full range of issues discussed above at once (Alphatech, Hughes). While all of these algorithms claim some level of performance, and all the contractors we have talked with have agreed that most of the issues noted above are critical and need to be addressed, we have yet to see any of the sensitivity studies necessary to compare two different algorithms or even to evaluate a single algorithm. Until detailed sensitivity studies using Monte-Carlo techniques and the error sources described here are available, the quality and timeliness of tracking information necessary for the strategic defense mission cannot be assured, and the effects of realistic tracking errors on the rest of battle management will remain unknown.

Sponsor: RADC

Contractor: Alphatech Inc.

Algorithm: Track-Oriented Hybrid State Multiobject Tracking Algorithm

Alphatech Incorporated's effort in multi-target tracking algorithm is quite comprehensive. Their algorithm includes many real-world sources of error and appears to be close to an implementable piece of code, rather than a low-fidelity computer simulation. Aside from the high fidelity, there are several other noteworthy features in their algorithm, as described below.

First, the algorithm is designed to be sensor- and target-model independent. Various sensor modules can be used to allow fusion of all types of multi-spectral observables, while various dynamic and measurement accuracy parameters can be included in different target models. Their tracking state vector carries this philosophy one step farther. Continuous-valued states model the standard position-velocity of the target, while discrete-valued states model target-sensor-measurement associations. These discrete states allow sensor-error types and characteristics to be included in the track, not just the sensor errors, as in conventional target-state Kalman filters.

Alphatech's algorithm also uses some unique approaches to solve the track-correlation problem which aid in both tracking and higher level battle-management functions. Many of the track-correlation algorithms we have seen allow for multiple associations from scan to scan. These associations result from multiple-target returns (or false returns due to clutter or noise) in the gate where the Kalman filter predicts that the target will appear in each successive scan. If the gate is large compared to the clutter/noise level or compared to the target density, many returns can occur within a gate, causing a geometric increase in the number of tracks that must be maintained. The tracks caused by these multiple associations form a "tree" which is eventually "pruned" by some optimal or heuristic rule set, so that eventually only the true track of an individual target remains.

Alphatech's approach is to use a number of target-kinematic models (from the target modules) in the Kalman filter, so that separate gates and resulting associations are made for each postulated target type. Although this may cause an initial increase in the number of tracks maintained (by a factor of the number of target models used), the modeling of the kinematics will be of a higher fidelity for the correct association/target model track than if a single target model were used. This higher fidelity enables more efficient pruning of incorrect track associations. The battle manager can then use the target model in high confidence tracks to perform missile-typing in boost phase, perform discrimination while tracking RV and decoy deployment from the bus, and perform active discrimination for methods which impart a ΔV to the target.

The second payoff for the battle manager results from the pruning process itself. If a track is given N scans to pass the pruning criteria before it is abandoned, the number of scans (1 to N) a track has survived is a measure of track confidence. In Alphatech's tracking algorithm, tracks are grouped according to their confidence level, providing the battle manager with an additional measure for engagability to use in an optimal weapon-to-target assignment algorithm.

The Alphatech tracking algorithm has been coded and verified on both single processor and parallel processor (hypercube) computers.

Sponsor: Mitre Corporation IR&D

Contractor: Mitre Corporation

Algorithm: Stereo-Optic Target Tracking and Sensor Fusion

The Mitre Corporation has proposed an algorithm which takes a geometrical approach to solving the sensor-to-sensor track correlation problem. The key to this algorithm is to establish a plane containing the target and two sensors observing the target. Target position is determined by the two angles in the sensor-sensor-target triangle, observed at the two sensors. The critical point is that the Mitre algorithm assumes that two major sources of error are negligible sensor-to-sensor alignment errors and angle errors associated with correlating sensor measurements in time (scan-to-scan correlation). Sources for sensor alignment errors are discussed in the Common Reference Frame (CRF) sections of this paper. This assumption is shared by many other such algorithms.

The scan-to-scan correlation problem is discussed in more detail here. The Mitre algorithm assumes that, at any instant, say t_b when sensor one scans a particular target, sensor two can provide the line of sight (LOS) angle between sensor one and the target. With scan times on the order of seconds, such interpolation could introduce significant errors in the angle measurements, errors which the Mitre algorithm does not consider. The errors and sources of error associated with these factors need further analysis.

Although Mitre claims its approach to sensor-to-sensor correlation can be used for scan-to-scan correlation, it is not clear how such an association would be performed.

Mitre's technique may be applicable to the two-sensor probe concept. Since only two sensors are involved, they may achieve sufficient alignment and the coordination necessary to perform sensor-to-sensor correlation as described by Mitre (although the scan-to-scan correlation problem remains unsolved).

Sponsor: AFSD

Contractor: Hughes Aircraft Company, Electro-Optical and Data Systems Group (EDSG)

Algorithm: Tracking Algorithms for SSTS

The EDSG at Hughes Aircraft Company has been deeply involved in the technology assessment (including algorithms) of SSTS. Our exposure to the work being done at EDSG is presently limited to two documents. The first is a book of course notes on Multiple Target Tracking taught by Dr. Drummond (the course was attended by several IDA/STD staff members in December 1986) which highlights the problem in general, with some mention of work being done specifically at EDSG. The second is an EDSG internal memo by Dr. Drummond on "Technology Assessment of SSTS Algorithm Development." We are awaiting a report written by EDSG for the Air Force Space Division which will contain several sections devoted to the algorithms EDSG is developing for SSTS.

Since we do not have a specific algorithm to discuss (at present), we can only describe in a general sense the work EDSG is doing in tracking algorithms. EDSG seems to have a firm handle on the SDI tracking problem, primarily because of their awareness of the real-world problems associated with implementing such a system. Whereas all other efforts we have seen to date have approached the tracking problem in a "top-down" analysis, EDSG is working the problem from the sensor system and subsystem perspective. In fact, many of the attributes listed under the track initiation and maintenance function come from the multi-target tracking course notes. Our visits to contractors and the recent SDI Advanced Computing Workshop confirm that Hughes, EDSG, and Alphatech are by far the leaders in high-fidelity tracking algorithm development that we have seen to date.

The following are key points from the Hughes memo on SSTS algorithm assessment:

- Before specific algorithms can be selected, trade-offs between performance and required resources must be evaluated.
- Signal-processing algorithms will greatly influence the design and performance of higher level sensor and tracking functions.
- Implementable algorithms have been fully developed, but not for the target numbers and densities that SSTS may encounter. Many high-level tracking algorithm concepts have been proposed, but performance estimates can only be made with massive fine-grain Monte Carlo simulations.
- Low-fidelity simulations of tracking algorithms may be misleading. The effects of including false signals in the sensor field of view should not be ignored or underestimated.
- Different algorithms may be necessary for each sensor mix and set of sensor characteristics being considered.
- Theoretical bounding of tracking errors using the Cramer-Rao criterion is possible for single-target tracking with no false signals. However, this and similar methodologies do not apply to the multiple-target case. Monte-Carlo simulations are required to analyze tracking errors for the multiple-target case. However, the criteria for evaluating multiple-target estimation performance have not been well established.
- The characteristics of the algorithm should be established prior to final sensor system design. The development of hardware and software cannot be properly undertaken before the algorithms are well understood.
- The following algorithms are resource intensive, and will require trade-offs between resources and performance, including track initiation, track maintenance, filter update, sensor handover. These algorithms are critical because there is a risk that they will not achieve the peformance desired of the system with respect to passive and interactive discrimination and maneuver detection. All of these algorithms are interdependnt to some degree, and cannot be evaluated in isolation.
- Primary features of interest to the critical algorithms are: resources required, target acquisition time, discrimination accuracy, number of false and missing tracks, estimation and prediction accuracy, estimation error covariance matrix realism (accuracy), situation assessment.
- Critical aspects of the threat and the algorithm which determine the importance of these features are: algorithm parameter adjustment, algorithm sensitivity to

faulty data, adaptation to system degradations, distribution of processing over many platforms, algorithm interaction, impact of misassociations, robustness, impact of clutter and nuclear events, efficient use of resources.

Sponsor: Army

Contractor: Nichols Research Corp. (NRC)

Algorithm: Multi-Sensor Correlation and Target State Estimation (MSCTSE)

The goals of the MSCTSE algorithm are to provide sensor-to-sensor correlations of single-sensor tracks. These tracks are formed on the sensors themselves, then are passed down to different battle-management nodes (depending on the sensor type) where the MSCTSE resides. The algorithm was described for the midcourse SSS sensor trio.

The scan-to-scan correlation is performed on board the associate sensors. Tracks are prescreened before the assignment process to weed out observation pairs that are widely spaced along the line-of-sight vector to the sensor. Surviving observation pairs that exceed some "closeness" threshold on the focal plane (by virtue of the tracking covariance error resulting from a forced pairing of the two observations) are dropped to further reduce the data-processing requirements of the assignment algorithm.

Several candidate algorithms are being used for the actual assignment process. The most promising candidate is a hybrid Bertsekas-Sparse Munkres algorithm. Communications and data processing requirements have been estimated.

NRC's approach to the multi-sensor/multi-target tracking problem is similar to others, in that the sensors themselves produce tracks with only their own observations, then hand over to the battle manager for fusion. The approach also seems to stress the importance of minimizing the computational load of the correlation algorithm; the two-stage filter to eliminate track pairs (trios) prior to correlation is unique. However, the process for setting the covariance filter threshold is not described, and the problem of scan-to-scan correlation onboard each sensor is not addressed and appears to have been written off as a sensor problem.

Sponsor: USAF

Contractor: Applied Technology Associates (ATA)

Algorithm: Tracker-correlator

The overall effort by ATA breaks down into four stages:

- Single sensor, single-object track initiation
- Single sensor, single-object track maintenance
- Single sensor, multiple-object track initiation and maintenance
- Multiple sensor, multiple-object track initiation and maintenance.

The first two functions were implemented in Phase 1 of the effort; Phase 2 will deal with the remaining two.

Track establishment, e.g., initiation, is implemented in this approach from two angle-only observations obtained by a single sensor platform. This is shown to be possible by using not only the angle data but also the time elapsed between measurements, and by restricting the family of candidate tracks to those associated with launch and aim points within the USSR and U.S.A., respectively. The launch and impact points of a nominal trajectory (10,000 km, minimum energy, liquid or solid booster, with altitude as a function of downrange distance) are fit to the two observations. ATA has found that variances between the actual and nominal trajectories induce sufficiently small errors in the state vector to allow its utilization for initializing the Kalman Filter.

The track-initiation algorithm is novel, and addresses an important step in the overall process of tracking threat objects. The accompanying analysis in ATA's reports is extensive and is supported by results from modeling to justify feasibility. An important portion of the analysis is devoted to rocket propulsion and orbital dynamics.

An important remaining question is whether the path taken will prove productive when extended to the more realistic multiple-sensor, multiple-object environment.

Sponsor: USAF

Contractor: PAR Technology Corp.

Algorithm: Boost-Phase Analysis (track establishment and maintenance)

The initial inputs to this algorithm are passive observations from two sensors during the boost phase. The output is ICBM position, velocity, and error covariance at burnout. A number of steps are invovled, as follows:

- ***Data preparation.*** The data acquired by the first sensor during an initial interval, say 30 to 60 s, is made to coincide in time with the data obtained from the second sensor, through a polynomial interpolation technique. Subsequently, launch time and location are estimated.

- ***Launch azimuth estimation.*** This is accomplished through the estimation of the intersection of lines-of-sight.
- ***Observation transformation.*** This is merely a coordinate transformation into the launch azimuth plane.
- ***Trajectory filtering.*** This is a Kalman filtering operation performed on the sequential observations from the two sensors.

The algorithm handles data from a single object. The problem of associating sequential observations from many objects with the proper grouping, i.e., correlation, is not considered here. It is not clear that in this real-life complex environment the proposed method of initialization and correlation yields satisfactory performance, or for that matter, is feasible.

Sponsor: Army

Contractor: Alphatech and Honeywell

Algorithm: Multiple Information Set Tracking Correlator (MISTC)

The overall construct consists of a set of coordinated algorithms for the following functions:

- Single object, single-sensor track establishment and maintenance.
- Single sensor, multiple-object track establishment; i.e., single sensor scan-to-scan correlation.
- Local track fusion; i.e., the fusion at a sensor of all the tracks relating to all the objects seen by that sensor, based on track reports from all other sensors which see those objects.
- Global track fusion based on a hierarchical structure of tracking nodes, local fusion-management nodes, and a global fusion-management node.

The work performed so far resulted in some valuable insights, approaches, and factors to be considered in any implementation, such as:

- A dynamic asynchronous communication and correlation methodology
- A redundant hierarchical sensor architecture
- A detailed characterization of the data to be transferred between nodes
- A mathematical model and track-forming algorithm based on a tree structure of track candidates and associated likelihood functions

Sponsor: U.S. Navy

Contractor: VERAC, Inc.

Algorithm: Tracker-Correlator Algorithm

This algorithm uses artificial intelligence (AI) techniques to fuse multiple observation types, and provide threat cloud RV and decoy parameters to the battle manager. The algorithm uses two expert systems, one containing rule sets for object characterization, the other with rule sets for payload partitioning of individual boosters. Sensor inputs to this algorithm include long-wave infrared (LWIR) and laser radar (LADAR) imaging and doppler data. The algorithm provides the battle manager with classifications of individual objects, and the associated probabilities that those classifications are correct.

C. DISCRIMINATION

1. Description

The general topic of discrimination is divided into two catagories: boost phase discrimination (or missile typing), and midcourse/terminal discrimination. In the general sense, both types of discrimination capabilities give the battle manager and the National Command Authorities a description of the threat which can be used to determine the best appropriate defense. On the general level, discrimination provides the necessary information for the mode-selection algorithms; at a more detailed level, discrimination provides a value estimate for each target so that the weapon-to-target assignment algorithms can execute an efficient engagement. Discrimination algorithms must perform the following functions:

- Fusion of observables
- Estimation of population statistics (estimating the mean and possible spread in measurements of the observables for each possible category of object: booster type, RV, decoy)
- Object classification and/or determination of object value

Many discrimination schemes require that staring sensors be sequenced between objects in midcourse in an optimal fashion so that a maximum number of objects can be scanned in a given time. Although vital to executing the discrimination process, such sensor scheduling algorithms are not classified as "discrimination" algorithms, and are therefore covered under "sensor resource management."

During boost phase, the missile typing function is to distinguish high-priority offensive threats (SS-18s) from relatively low-priority threats (SS-25s) and to identify ASATs. Since this is the phase in which enemy RVs can be destroyed most efficiently, it is essential that missile typing be completed accurately and quickly so that each booster's value is known in time for it to be engaged before it deploys its RVs and decoys. Missile typing can be accomplished in many ways. If a booster type can be associated with a specific launch site, a booster track can be extrapolated back to the launch site to identify it. The plume signature characteristics and/or fly-out profile can be compared to known data to obtain an estimate of the missile type. Therefore, the missile typing process is primarily one of comparing what is measured to what is known about each type of booster.

During and after the post-boost phase, the discrimination function distinguishes between RVs and decoys. It is generally assumed that intelligence on enemy boosters is significantly better than intelligence on enemy decoys, therefore, the *a priori* data used for comparison with measurements of boosters in the boost phase will not be available for comparison of RVs and decoys for the remainder of the battle (or at least the estimates on decoy characteristics will be far less certain than the booster estimates). It follows that the statistics of the observables of RVs and decoys must be estimated from each observation sample. For example, if accurate missile typing is available, the throw weight of a given booster might be known, but its off-loading of RVs for decoys will not. Therefore, even the number of RVs in a booster-load of objects must be estimated from discrimination measurements. If the first midcourse discrimination measurements are made after several booster-loads have merged to form a single threat cloud and the battle manager does not know how many boosters contributed to that cloud, the missile typing information (i.e., throw weight) cannot even be used to place an upper bound on the number of RVs in the cloud. Furthermore, while in concept missile typing is feasible, a demonstration of performance is highly desirable.

Many types of sensors from many different viewing angles may be used for discrimination including Long Wavelength Infrared (LWIR), Shortwave Infrared (SWIR), Neutral Particle Beam (NPB) returns and Laser Doppler Radar (LADAR), and terminal phase slowdown. Since information from these various sources must be combined, one of two approaches to the discrimination process can be used. First, the estimation process described above can be multi-dimensional (one dimension for each measurement type) so that a state-space is defined with one area populated mostly by RVs and another mostly by decoys. Otherwise, the various measurements can be optimally combined into a single quantity which is used as a discriminant. If the measurements have a Gaussian distribution, the first approach always can be simplified to the second.

As the battle progresses, there may be several opportunities to acquire discrimination measurements from a threat cloud. If the discrimination phenomenology involves transients (object change in temperature as it passes through the terminator), or if measurement error decreases with the number of observations made, discrimination capability will increase with time. Therefore, the data base used for parameter estimation of the threat cloud must be continually updated, so that each additional measurement is used most effectively.

The final discrimination process uses the estimates to place a threat-value on each object in the threat cloud. One approach is to set some type of threshold appropriate to the estimate statistics and the defense's resources so that each object is either classified as an RV or a decoy. Another approach is simply to give each object a value equal to the probability that that object is an RV. The choice of a binary or continuous-value structure (or anything in between) is dependent on the other battle-management algorithms (mode selection, weapon-to-target assignment) that will be using this value information. Because it is based on more detailed data, the first approach--all other things being equal--would yield better results.

2. Critical Features

a. Factors

- ***Quality of booster intelligence estimates.*** Since boost-phase discrimination will depend largely on comparison of intelligence estimates with observed booster phenomena, errors in these estimates will produce errors in booster typing.
- ***Number of different measurement types.*** As described above, the complexity of a discriminator algorithm is dependent on how many types of measurements must be combined, and therefore, how many parameters (RV and decoy statistics for each measurement type) must be estimated.
- ***Discrimination capability over time.*** If appropriate, the discrimination algorithm must be able to combine the measurements from many observations taken over time.
- ***Environmental effects on measurements.*** If measurements from many objects are to be compared for estimation purposes, and the environment affects one group of objects differently than another, the contribution of these

effects must be subtracted from the measurements so that the statistics from one group correspond to those of another group. For example, if the temperatures (as indicated by LWIR measurements) of the objects in a threat cloud are to be used as a discriminant, all the environmental factors which could affect the observed temperature must be considered, along with how these factors might affect each object differently because of its location or aspect angle to the sensor. Some such effects are the proximity of an object in the focal plane of the sensor to the sun, the earth, or nuclear detonations.

- ***Sensor calibration errors.*** If multiple sensors are used to compare measurements between RVs and decoys, calibration errors between the sensors will prevent accurate comparison of the measurements.

b. Assumptions

- ***A priori knowledge of booster/RV/decoy phenomena.*** Observable booster/RV/decoy characteristics of each booster type are assumed to be known with certainty. This reduces the discrimination problem to one of pattern matching the observed data with the known data.

- ***A priori knowledge of measurement statistical parameters.*** If the noise characteristics and mean values of RVs and decoys for each measurement type are known, estimation of these parameters is not necessary. Since the accuracy of parameter estimation increases with sample size, estimates for a threat cloud consisting of only a few tens or hundreds of objects (as might be seen in the terminal phase) will be very poor, making the discrimination problem even worse. Including the *a priori* knowledge eliminates this problem, and makes the discrimination capability with a small threat cloud just as good as with a large threat cloud.

- ***A priori knowledge of the number of populations present in a discrimination sample.*** Many types of objects with different observable signatures (e.g., small-yield RVs, high-yield RVs, heavy decoys, balloons) may be present in a threat cloud, and it is desirable to classify each object as being from one of these populations. If the number of populations is known, then it does not have to be estimated from the data, making discrimination more accurate (see assumption above).

c. Figures of merit

- ***Operating characteristic.*** All decision processes can be characterized by an operating curve which plots two types of errors; probability of misclassification (e.g., decoy for RV) vs. miss probability (e.g., RV for decoy). For a given value of the first type of error, the second type of error can be used as a figure of merit.

- ***Least-squares curve fitting.*** Given a histogram representing the sum of the distributions of many classes of objects, derive the individual distributions by performing least-squares curve fitting to the points of the histogram.
- ***Maximum likelihood estimation.*** Using maximum likelihood techniques to derive the means and spreads of each population present in a sample of discrimination measurements from a threat cloud of objects.

3. Review of Algorithm Development Activity

We have yet to see any formal work in the area of algorithms for boost-phase discrimination (missile typing). Considerable effort has been devoted to midcourse/terminal discrimination, but we have seen only one approach to the estimation problem (IDA). Most of the work has concentrated on the observable fusion and classification-determination problems. We have seen no formal effort to include environmental effects. All the algorithms assume knowledge of measurement statistical parameters and number of populations present in a discrimination sample. From our investigations into the effects of estimation errors on discrimination capabilites, we strongly advise that the contractors begin to address this problem, as it seriously affects all the assessment functions of battle management and architecture performance estimates.

Sponsor: *AFSD*

Contractor: The Aerospace Corporation

Algorithm: Discrimination algorithm

The Aerospace Corporation has several groups working in the area of discrimination algorithms. All of their approaches are concentrated on the derivation of a one-dimensional K factor from multi-dimensional, time-varying IR measurements. The actual data they are using comes from Nichols Research Corporation's Optical Signatures Code or from a similar, less complex in-house code. Both of these codes simulate the multi-spectral returns from RV and decoy models given target, sensor, and environment parameters. Aerospace is developing ways to combine this information into a one-dimensional rank ordering of targets, from high-probability RVs to low-probability RVs. One group is going a step farther in developing heuristic algorithms which calculate linear and quadratic thresholding functions to classify objects as RVs or decoys.

This transition from multi-measurement observations to rank ordering and value determination of targets is critical for the discrimination function. The classification rules are less useful to optimal weapon-to-target assignment algorithms, since they require a higher resolution for RVs than the one bit RV-or-decoy classification. Aerospace can complete their discrimination algorithms by including parameter estimation with their fusion measurement algorithms. Their work on classification techniques would benefit from discussions with others developing optimal weapon allocation algorithms.

Sponsor: SDIO

Contractor: Command Systems Group (CSG)

Algorithm: Contact Discrimination Algorithm

CSG Inc. is developing a discrimination algorithm which uses artificial intelligence (AI) to fuse multiple observation types, and provide threat cloud RV and decoy parameters to the battle manager. The algorithm uses two expert systems, one containing rule sets for object characterization, the other with rule sets for payload partitioning of individual boosters. Though we have only seen the approach described to date, CSG intends to perform various sensitivity studies including algorithm performance sensitivity to threat design, sensor design, optimal battle management, penetration concept, decision criteria tuning. Since this algorithm is at such a preliminary stage, it is difficult to comment on the approach.

Sponsor: Army

Contractor: Nichols Research Corp. (NRC)

Algorithm: Multi-Sensor Discrimination and Classification (MSDC)

The goal of the MSDC is to process object data from multiple sensors to determine whether an object is a lethal threat and may additionally classify objects into one of nine distinct groups with corresponding probabilities which are used in the Threat Assessment (TA) algorithm.

Once a set of discriminants (three-color LWIR, LADAR imaging, etc) has been evaluated, it is compared to a preengagement Gaussian model for classification using a Bayesian approach. Each discriminant's contribution to the classification probability is adjusted by a set of *a priori* weights for each discriminant type. Multiple probabilites from multiple sensors are fused, two at a time, by a recursive process using Bayes Formulation. Communications and memory requirements have been derived.

The idea of weighting various discriminants to derive an "optimum K-factor" is common to many discrimination algorithms. However, since this is part of an integrated algorithm package, it seems that much information is wasted using *a priori* weights. For instance, the Regional Sensor Resource Manager (RSRM) has already derived orthogonality estimates which could be useful in correlating LADAR images of objects. The Battle Assessment (BA) algorithm could provide information on nuclear backgrounds which would affect the weighting of different LWIR bands. Since this type of information is available in the system, it seems that discriminator performance would benefit from a dynamic set of weights.

This algorithm is too dependent on *a priori* assumptions about the threat, such as (1) the discriminant means and variances for each discriminant type of each object type, and (2) the numbers of each object type present. Threat cloud estimation techniques could be used rather than relying totally on intelligence estimates of the threat. The algorithm will work even if there are errors in the values assumed for the parameters, albeit the performance will be degraded.

Sponsor: SDIO

Contractor: Institute for Defense Analyses (IDA)

Algorithm: Maximum Likelihood Threat Cloud Estimator

IDA initiated the development of this algorithm in order to quantify the errors involved in estimating a cloud composition. This algorithm uses a maximum likelihood technique to derive estimates on the number of RVs, K factor, and standard deviation from a sample of one-dimensional discrimination measurements, where there is one RV and one decoy distribution, and the statistics are assumed to be Gaussian. Sensitivity studies for sample size, actual K factor, and decoy-to-RV ratio have been performed for errors in the estimated number of RVs, estimated K factor, and estimated standard deviation. The basic algorithm can be extended to include more than two distributions of objects and non-Gaussian statistics. Cramer-Rao lower bounds on estimator accuracy have been calculated for comparison.

Sponsor: SDIO

Contractor: IDA

Algorithm: Sequential-Decision Algorithm

In the design of discriminator architectures, the same performance may be achieved by many small-K discriminators or a smaller number with large-K discriminators. "Small-K" and "large-K" in this context mean discriminators with smaller or larger K factors achieved, for example, through smaller or larger apertures or transmitter power. Furthermore, if more than one type of discriminator, e.g., imaging and NPB, is used, the number of each can be traded off to achieve equivalent levels of overall K factor.

If an object is examined more than once for discrimination purposes, the readings can be combined and a K factor achieved which is larger than any of the contributors. The composite K factor is given by:

$$K = \left(\sum K_i^2 \right)^{1/2} .$$

The expression allows for combining data from different discriminators to get a single random variable on which the decision is made.

Some interesting design approaches can be obtained by this method. For example, in NPB discriminators, since both the mean and the variance of the data are proportional to the sensor aperture, if we use N identical discriminators of aperture A, K would be proportional to $(NA)^{1/2}$ or $A_T{}^{1/2}$, where A_T is the total aperture area in all the discriminators used. This assumes that all these discriminators operate under the same conditions, e.g., at the same range, in which case the desired K factor is achieved through an A_T that can be subdivided through any number of devices. In reality, the K factor is closely range-dependent, so having many small apertures in space is highly advantageous because those close to the object will strongly boost the K factor.

The method outlined above for combining successive readings and thereby improving the K factor, opens the door to sequential-decision methods. Sequential-decision theory is particularly efficient when there are limited resources for making many decisions. In its original form, the theory prescribes that data on any one object is accumulated until the aggregate data indicates with some level of certainty that one of two hypotheses is true. Thus, the dwell time is dynamically variable from object to object, but the expected time expended to scan all the objects is less than the time needed for the same level of performance when the dwell time is fixed in advance.

In the case of discriminators, the method must be modified because of round-trip propagation delays. Conceptually, a group of objects is scanned for relatively short dwell times, the group is revisited, and data from revisits are aggregated. When the aggregate data from an object falls above one threshold or below another, lower threshold, a decision is made, and the object is eliminated from further revisits. As time elapses, objects are dropped and the revisit time for the remaining objects shortens because only those which are still in doubt are revisited. The method is particularly useful when both heavy and light decoys are present.

D. COMMON REFERENCE-FRAME (CRF)

1. Description

The various component nodes of the defensive system exchange data about their own positions and the positions of threat objects relative to themselves. Since most of these objects are continuously changing position, the data must carry time tags. These operational requirements necessitate the establishment of a coordinate system in which the instantaneous position of each object can be uniquely defined. It also necessitates the establishment of a system clock with which clocks at each platform can be synchronized.

To maintain synchronization, each platform must perform a number of operations. When a platform first enters the system, it establishes time and spatial synchronization with other platforms. This is often done by tracking the time-varying propagation delay. For example, the platform might transmit a code and measure the time it takes to receive it upon retransmission by the other platform. A continuous time and range measurement for several platforms yields unambiguous position information. Alternatively, a system-wide position-determination system can be used. A system such as the Global Positioning System (GPS) could be considered. Of course, the survivability and vulnerability of such a system is a major consideration.

2. Critical Features

a. Factors

- ***Nominal trajectories of the sensor and weapon platforms.*** The trajectories of the various platforms to be synchronized affect the resultant accuracies. Position updates must be sufficiently accurate to make any displacement between updates negligible. An alternative to high update rate is position prediction between updates.
- ***Accuracy and availability of attitude references.*** Line of sight to the attitude reference may not be continuous. Sensor resolution, propagation errors, system noise, and timing errors will all degrade position accuracy.
- ***Accuracy and availability of position references.*** Line of sight to the position reference may not be continuous. Propagation errors, system noise,

and timing errors will all degrade position accuracy. Whether these are disruptive depends on the application. For example, communication levels require synchronization to within one symbol, hence high-data-rate levels have stringent timing requirements.

- ***Accuracy and availability of a system clock.*** All systems require some kind of common time frame. Some applications, such as peacetime maintenance, require little timing accuracy; a maximum error of a few seconds can be achieved with an open loop clock which is reset from the ground once a day. Others require very high timing accuracy; e.g., when combining scans from more than one angle-only sensor to position an object in three-dimensional space, individual scans may not take place at the same time. Therefore, accurate interpolations with respect to time are necessary to coordinate data from two or more sensors.
- ***The choice of an appropriate frame in different phases of the battle.*** Different phases of the battle may require different frames of reference. Choosing an appropriate frame can considerably reduce the computational load. For example, a terminal engagement algorithm might use an earth-centered coordinate system with geographic coordinates; conversely, a space-based engagement algorithm might use an inertial frame of reference.
- ***Platform maneuvers.*** If the platform maneuvers out of its nominal trajectory to evade an ASAT attack, is the attitude/position referencing accuracy degraded? Must other defense elements be assigned because of the maneuver? Can small, routine maneuvers be accommodated?
- ***Communication capability to support the maintenance of a common reference frame.*** Time and position synchronization do not require high-data-rate channel. Nevertheless it is vital to have the proper connectivity to support it. Since often the communication requirements, e.g., the number of antennae and pointing requirements are developed on the basis of other requirements, such as high-data-rate critical links, it is important to ascertain that these connectivities can support the operations considered here.
- ***Inertial system characteristics.*** If attitude/position fixes are obtained at a low rate, an on-board inertial guidance system may be used to provide position and attitude information between fixes. Such systems are prone to biases and oscillations which should be taken into account in a common reference frame algorithm.

b. Assumptions

- ***Attitude references are exact.***
- ***Position references are exact.***

- ***Time references are exact.*** This would involve ignoring stability of the clocks (long and short term), propagation anomalies (which affect many operational systems such as Loran-C and Omega), and clock update rates.
- ***Constant availability of attitude and position references.*** Updates are assumed to be constant (or at least made at a high rate compared to the rate of error propagation) and uninterrupted.
- ***Sufficient communications for the position-determination task.*** Even if a position-determination system is available, from which the platforms can derive their position with sufficient accuracy, there remains the problem of distributing this information to all other platforms in a timely manner. Hence, assumptions about the characteristics of such a distribution system are relevant.
- ***Perfect inertial system.*** No position or attitude errors accumulate between attitude/position fixes.

c. Figures of Merit

- ***Accuracy.***
- ***Vulnerability.***
- ***Reliability.***
- ***Flexibility.***

d. Computational Methods

- ***All elements reference themselves to a primary element.*** Specific primary platforms in a constellation receive attitude/position references from the ground or other sources, then the other platforms in the constellation reference themselves to the primary platforms.
- ***An external positioning system is employed.*** The GPS or a ground-based equivalent is used by each platform.
- ***Stellar attitude reference.*** Each platform determines its own attitude with respect to specific reference stars.
- ***Centralized clock.*** One master clock is the time reference to which all other platforms synchronize through communications channels.
- ***Decentralized clock.*** Each platform or group of platforms carries a master clock synchronized through communications channels.

3. Review of Common-Reference-Frame Algorithms

Common-reference-frame algorithms have received scant attention within the SDI community. In all system-concept descriptions, as well as in the modeling activities, it is assumed that when two platforms have the same values for the coordinates of an object, the location of that object is identical for both platforms. This assumption implies a perfect common reference frame in which the positions of the platforms exchanging data are known with zero error. Also, since we are dealing with moving objects, there is the implication of perfectly synchronized, errorless clocks. As a rule, these assumptions are not stated explicitly. It is not the function of this study to analyze the validity of these assumptions; as a minimum, one would like to see an evaluation of the errors injected into various functions by some reasonable postulated common reference frame implementation. For example, given the update rate of the on-board clocks and the clock stabilities, one could compute the position error involved in guiding the interceptors. This, in turn, would lead to the requirements for clock update rate.

The magnitude of the errors depends on the method used for implementing the common reference system. Past work on synchronizing a network of many platforms is of interest and some of it might be transferable. Synchronization of mobile satellite communication terminals, e.g., in Time Division Multiple Access or in spread-spectrum systems (where timing accuracy is essential), are two applications worth examining. Other system concepts with similar requirements are data networks involving mobile tactical units, e.g., past work on the Joint Tactical Information Distribution System (JTIDS).

E. THREAT ASSESSMENT

1. Description

While the role of discrimination algorithms is to determine the classification or value of each object in the threat, the role of the threat-assessment algorithms is to determine the overall structure of the threat. By taking inputs from the discrimination and tracking algorithms, the threat-assessment algorithms estimate the magnitude of each part of the threat and how they can affect defense elements as well as defended assets. These threat-structure estimates are then used by the mode-selection algorithms to determine the best course of action for the defense.

In boost phase, the threat assessment function's first concern is estimating damage to the defense by boosters typed as ASATs. Sensor and weapon platforms in danger of being intercepted would have to be identified, and the effects of their loss on the defense in the face of the current or predicted threat would have to be estimated. The threat-assessment algorithms would perform the necessary trade-offs for the number of interceptors allocated for self defense versus the number used for engaging ICBMs. These algorithms will influence the values that targets are assigned in the boost phase, based on the observed mix of ASATs and ICBMs.

The threat-assessment algorithms must also be concerned with the type of attack. A full-scale Soviet attack would require one defense posture, while a limited attack from a third-world nation or terrorists would require another. These algorithms must also be concerned with comparing the observed threat to intelligence estimates of total enemy capacity so that some weapons may be reserved for a second attack wave or a prolonged attack.

As the battle moves into the midcourse and terminal phases, tracking algorithms will provide ever-improving estimates of the impact points for each threat object. As impact point prediction becomes available, the threat-assessment algorithms will compare the threat concentration on an impact point, the hardness of assets at the impact point, and the inventory of available point defenders at the impact point. Threat structure must also be identified since a "string" of RVs falling on a site could detonate one at a time at decreasing increments in altitude, blinding point defense sensors and interceptors, and virtually guaranteeing that the final one or few RVs would land on target. If a string is identified, a high engagement value can be placed on key members of the string so that the attack is "destructured." If a highly concentrated attack is launched, the threat-assessment algorithms must be able to recognize impact points which stand little chance of survival so that interceptors normally allocated for their defense can be used more effectively elsewhere. Therefore, in terminal phase as well as in boost phase, the threat-assessment algorithms will influence the values assigned to targets.

2. Critical Features

a. Factors

- ***Carrier vehicle necessity after SBI launch.*** If a CV is required to send guidance updates to all its SBIs in flight, it will be necessary to defend the CV until the engagement of its final SBI, not just until all of its SBIs are launched.

- ***ASAT kill capabilities vs. defense platform survivability.*** If an ASAT is to be effectively killed, it must be engaged successfully before any defense platform lies within its kill radius.
- ***Impact point hardness/value vs. time.*** Ports and airbases may need defending only until the ships, submarines, bombers, etc., that are based there can be launched.
- ***Potential scenario list.*** The observed threat should be compared to a list of anticipated or potential scenarios so that a rapid and detailed threat structure evaluation can be provided to the national command authorities and the mode-selection algorithms. Such scenarios would include threat statistics based on intelligence data, along with the nation where the threat originated.

b. Assumptions

Because there has been little effort in this area, a list of assumptions cannot be made at this time.

c. Figures of Merit

Again, because there has been little effort in this area, a list of possible figures of merit which a threat assessment algorithm might minimize or maximize cannot be made at this time.

d. Computational Method

We foresee only heuristic-type algorithms being applied to this problem, such as the example we have listed below.

3. Review of Algorithm Development Activity

Reviewing algorithm development is critical, but has not received the attention it deserves. In many of the analyses the threat is generally simple and canonical, with no "surprises" for the defense. We suggest that more technical Red-Team input be provided to the contractors so that they have some realistic threat structures to analyze.

Sponsor: Army

Contractor: Sparta

Algorithm: Threat Assessment (TA) for terminal phase

The goals of the TA algorithm are to recognize threat structure, estimate target impact point, predict asset survival, and provide this information in an appropriate format to the planning and assignment algorithms. Some important features of this algorithm are listed below:

- Nine-category probability classification; three types of RVs, three types of decoys, and three types of debris
- Impact point prediction with error footprint based on tracking covariance
- Blast radii for each RV type
- Cumulative damage for ground assets of varying hardness based on distance from predicted impact and probability of survival
- RV "string" structured attacks on a single target area which could blind defenses with successive nuclear detonations
- Structured attacks which concentrate RVs to overwhelm HEDI and LEDI defenses
- Time-sensitive assets such as Strategic Air Command (SAC) airfields of submarine bases which must be defended at least long enough to launch retaliatory strike.

Sparta's approach is the only systematic approach to the threat assessment problem (in terminal phase) that we have seen to date. The detailed information above may or may not be useful to the battle manager because the nine-category classification used to estimate high-, medium-, and low-yield RVs to calculate blast radii, etc., may be too detailed for the quality of available discrimination data.

F. INTERCEPTOR-TO-TARGET ASSIGNMENT ALGORITHMS

1. Discussion

a. Overview

Before reviewing the status of interceptor-to-target assignment algorithms, several issues which currently confront the SDIO should be discussed. These issues arise from the demands placed on the SDIO for early deployment, while systems appropriate to the missions required in the far term are still developing.

For the early phase of deployment, the task of interceptor-to-target assignment will not be a "show stopper." In an interceptor-poor architecture, almost any reasonably efficient method of pairing interceptors and targets will suffice. It does not matter if 60 percent or 70 percent of the RVs leak, because the mission of the early phase is only to introduce some degree of uncertainty to the Soviet attack planner. Thus, in the near term, an extremely sophisticated algorithm is not essential to the success of the mission.

However, in the later phases, the mission becomes one of a more robust defense, with a high level of confidence in the protection of certain target classes. Here, sophisticated algorithms are necessary. If the defense can destroy several hundred more RVs simply by using a better algorithm, it may be well worth the price of extra computing capability, given the cost in Space-Based Interceptors (SBIs) needed to achieve the same increase in defense.

Progress is being made in the development of algorithms for the allocation of interceptors to targets. Existing techniques and newly-developed algorithms are being used to solve the assignment problem in any number of formulations. Initially, most efforts focused on solving the boost/post-boost problem, but recent efforts are being aimed toward the terminal and midcourse problems. Strategies for adaptive midcourse and terminal defenses are being developed. One area which appears to require more effort, however, is the development of strategies to defend against an attack of ASAT weapons.

A second issue arises because the DAB Architecture is not fully defined, and algorithms for it cannot be developed. Granted, some of the algorithms may serve to define the architecture, but the process also works in the other direction. For example, until the midcourse sensor is decided upon, a firing algorithm cannot be fully developed for midcourse, because of the radically different timelines that result when a probe is used instead of an SSTS.

Similarly, some of the interceptor developers take the position that algorithms for an interceptor with high PK will suffice in this system. This eliminates the need for development of algorithms incorporating such features as salvos and shoot-look-shoot strategies. We caution against development of only these types of algorithms. Until it has been proven that the PK of the interceptor is near unity, we suggest that all algorithms being developed be capable of incorporating these features.

The SDIO should also focus on how battle information is passed from one phase to another during various phases of deployment. In particular, if the terminal phase battle manager must rely on the ground-based radar to reacquire all objects upon reentry, the algorithm used by the battle manager will reflect the fact that it does not know what objects are still in the midcourse phase and will soon be arriving. Conversely, if the terminal battle manager is receiving constant updates on all objects in flight, the algorithm will be one with a method for reserving interceptors for RVs that will soon arrive.

Issues such as those described above must be resolved before any sophisticated algorithms can be developed for the interceptor-to-target assignment problem.

Like many other-battle management problems described in this report, the weapon-assignment problem requires a very different solution technique in the early phases of the battle from that in the later phases. This is a direct result of the fact that the goals of the defense and the information available to the battle manager change from phase to phase.

In the boost and post-boost phases, the primary goal of the defense is to destroy as many RVs as possible. This results in the number of RVs on the missile at intercept being the primary factor in determining the value of the intercept for any interceptor-target pairing. Needless to say, this can be degraded by the expected PK of the intercept. Likewise, any number of other features may enter the value equation, such as whether a weapon platform is entering or leaving the battle space, the number of weapons remaining on a platform, or the expected time of flight, but it is still expected that the primary factor determining the value of an interceptor-target pairing will be the number of RVs on the missile (bus) at the time of intercept. This value will be independent of how many other targets are in the threat and how many other weapons are available to intercept the threat. The important point, in a mathematical sense, is that in problems such as these, the goal of the defense will usually be formulated in terms of maximizing the number of RVs destroyed, i.e., the value is placed upon the target.

Conversely, in the midcourse and terminal phases, the defense has considerably more information in the form of predicted aimpoint. Also, the defense's mission will now be one of preserving some fraction of each class of aimpoints, and thus the strategy becomes far more complicated. In these phases, a value cannot be placed on each RV until the defense assesses how many RVs are going to each target. For example, for two equal-value aimpoints, the defense should choose to preferentially defend the more lightly attacked aimpoint, if it has limited weaponry. So in problems such as these, the defense must usually formulate its problems in terms of maximizing aimpoints (or aimpoint value) to be saved, i.e., the value is now placed upon the assets to be saved, and this value is dependent on the number of RVs aimed toward it.

The interceptor-to-target assignment problem must be solved in a rapid, reliable manner if the system is to perform effectively. The importance of these algorithms and their utility in all phases can be illustrated by examining the battle scenario. At missile launch, the space-based sensors relay information about the targets to the battle manager. Regardless of where and how the tracks are formed and the missiles are typed, the allocator must decide which interceptors to assign to which targets. The decision will be based upon some value assigned to the targets and this value might depend upon such things as missile type, length of the engagement window, quality of the track being formed, and the expected PK for the engagement. As of this writing, algorithms and techniques for assigning interceptors in this phase are the most advanced.

As the missiles move into the midcourse phase, other issues will complicate the decision process, namely the presence of decoys, the quality of the impact-point prediction of the defense, the values of the defended assets, and the effects of nuclear bursts on sensors and communications links. The battle manager working in a target-rich enviroment must narrow its set of credible targets based upon the information it receives from the discriminants. Meanwhile, it must decide on how good its impact-point prediction is and how quickly the prediction is improving in deciding whether to hold or commit interceptors. The battle manager must also fight the battle in a manner consistent with prescribed goals, such as saving a fraction of the U.S. ICBM force or a fraction of the population centers. The entire process will be complicated by redout and communications disruptions resulting from nuclear bursts. We have yet to see interceptor assignment in a cluttered midcourse environment treated in a manner appropriate to the level of complexity which the actual defense will face. This is a difficult problem, given the underlying communications and environmental constraints. Considerably more effort should be devoted to resolving this problem.

Aside from the problems discussed above, the terminal-phase battle manager faces unique problems because this is the last line of defense. The interceptor allocator must constantly consider trade-offs of current shots at somewhat credible targets against future shots at RVs in later waves. This problem has yet to be adequately addressed. Virtually every model which attempts to solve this problem makes the assumption of separate aimpoints which cannot cross-defend each other, or if cross defense is allowed, it is not done with a mission in mind, but in a subtractive fashion.

Most of the effort to date in developing algorithms for allocating interceptors has focused on the early phases of the battle--the boost and post-boost phases. Very little, if any, effort has been devoted to solving the problem of allocating SBIs against a threat including ASATs and ICBMs. Most models which simulate this battle contain very inefficient battle managers. The state of the art in developing algorithms for midcourse battle managers is also lagging.

Selection of algorithms to be developed and studied must be consistent with the rest of the BM/C^3 subsystem. For example, an algorithm requiring a totally centralized battle manager must be supported by a communications system with highly efficient and rapid communications abilities. Likewise, an ASAT strategy allocating SBIs and ICBMs preferentially over incoming ASATs will require a system with extremely accurate track handover so that CVs can "fire and forget." We encourage the SDIO to enforce real-world constraints on algorithm developers so that the resulting solutions can be implemented by the system, and do not represent just nice solutions to abstract problems.

A brief review of the interceptor-to-target-assignment algorithm work by SDIO contractors revealed that the work falls into two categories. The first is the development of algorithms for incorporation in models used in studies of candidate architectures. The second is the development of sophisticated algorithms intended to function in the real system. In the first category, the algorithms are generally simplistic and of little use to the BM/C^3 Directorate. These are briefly discussed in the section on algorithms developed for the Strategic Architectures Office. However, several efforts in the latter category have merit and hold promise for further development and are discussed later (Section 3).

b. Conclusions and Recommendations

From discussions with various researchers at the companies above, we come to the following conclusions and recommendations about the algorithm development (aside from those discussed earlier).

- The SDIO is far from having any real assignment algorithm in hand for an initial deployment. The techniques were all designed to work well against a massive, spike launch, but will not necessarily provide a robust defense against staggered or small launches. The concept of robustness in all scenarios must be stressed to the algorithm developers.
- It is not clear what is should be "optimized" in these algorithms. Some approaches attempt to maximize RV destruction while others attempt to optimize CMT survival. Detailed figures of merit for the system are needed before any final development of algorithms can occur.

- The concept of "mode selection" and its effect on the algorithms has not been investigated and merits further study.
- For any algorithm to be implemented, some type of clustering will probably have to be performed by the defense. As described later in this section, both TRW and Alphatech have developed algorithms that rely on the defense clustering into battle groups and the offense being clustered into target groups. Clustering algorithms merit further study and their effects on overall system performance should be examined.

2. Critical Features

In this section, features of the interceptor-to-target assignment problem are discussed in detail, along with the commonly used methods for solving these problems. Factors which should be considered when defining the problem are discussed along with common assumptions and their validity. Common figures of merit for both the solution to these problems and the appropriate algorithms are also discussed.

a. Factors

The problem definition for assigning interceptors to targets will require information on many aspects of the battle, including the following:

- ***Spatial and time distribution of the threat.*** The allocation used by the defense should take into account the time and space concentrations of the threat. The algorithm used to solve the problem should be flexible enough that if the threat is not concentrated, the algorithm will take advantage of this and possibly salvo when interceptors are available in the boost phase. Conversely, if the threat is concentrated, the algorithm should recognize a massive attack and do all that is necessary to thin it out. Likewise, in the terminal phase, if a single RV is coming toward a target, the defense may choose to salvo interceptors at it, but if many RVs are incoming, the algorithm may cause a single interceptor to be assigned to each credible target. Until the handover capability of the defense is better defined, algorithms for the terminal phase cannot be refined.
- ***Spatial and time distribution of defensive interceptors.*** Any reasonable description of the problem would take into account the remaining time an interceptor will be in the battle space and ensure that it is used before the engagement window closes. Aside from the obvious case of the Space-Based Carrier Vehicles (SBCVs) which leave the battle space due to orbital motion, this aspect of the battle must also be considered for the ground-based interceptors whose footprints constantly shrink in time.

- ***Weapon system characteristics.*** The ideal formulation of the interceptor-to-target assignment problem will take into account the characteristics of the interceptors themselves. Simply having a nearby target does not guarantee a successful intercept. The background (either salvage fuzing or bright earth) can render an intercept impossible and the algorithm used must take this into account. Likewise, a desirable allocation algorithm is one which selects the intercepts with the highest possible return, based upon target value and expected PK of that particular intercept. Finally, the problem should be set up so that the defense is adaptive. That is, if the PK is near unity, the system would not salvo, but if the PK is low, salvos would be fired whenever sufficient inventory exists.

- ***Environmental effects.*** The effects of external events, e.g., nuclear bursts, should be included in any assignment algorithm. For example, intercepts can be planned so that salvage fuzing effects from one event will not have an effect on subsequent intercepts. Likewise, feasible intercepts may have to be rejected based upon predicted background conditions (e.g., cold-body seeker looking down at the earth).

- ***Frequency and quality of discrimination information.*** Whatever the method used in the midcourse phase for allocating interceptors (SBIs and ERIS) to targets, the defense will rely heavily on the information received from the discriminants. For example, if the discrimination information is sent once and never updated, the defense can choose to begin firing immediately after receiving all of the information. If the discrimination information trickles in over several hundred seconds, an algorithm which can begin firing on the basis of limited information is appropriate. Conversely, if the discrimination information is being constantly updated, an algorithm which will trade lost battle time for improved discrimination is needed.

  The discrimination can reach the allocator in two ways. All objects can be assigned a "yes/no" rating as to whether or not they are credible threats. A more sophisticated method is to rank the objects in terms of the likelihood of their being RVs. This latter method allows the allocator to set the discrimination thresholds such that the targets most likely to be RVs will be fired on, thus allowing the defense to pursue a more efficient firing doctrine making use of salvos toward very probable RV targets.

- ***Real-time reassessment of target values.*** The defense must be able to handle real-time updates on target values and the formulation of the problem should include this feature. For example, if a ground site has been destroyed, this information should be relayed to the battle manager, and the algorithm should be able to change priorities on the RVs already in flight. (ADD in the Martin work on redirecting SBIs in flight.)

- ***Concession of aimpoints.*** The formulation of the problem must allow the defense to "decide" when an aimpoint is too heavily attacked to defend it. Otherwise, the defense may continue to expend interceptors at RVs going to a heavily attacked target, when there is no hope of saving it.

- ***Defining a minimum return per expended interceptor.*** If the problem is well formulated, the defense will set an allowable minimum expected return. If the return drops below the established level, interceptors will not be fired. This will prevent the defense from salvoing an unlimited number of interceptors at any one target. To have a system with this feature in its problem formulation, it is assumed that a global battle manager exists.

b. Assumptions

In the work examined so far, many simplifying assumptions have been made and many of them are examined here. An assessment of the error introduced by the assumptions is also included.

- ***Perfect Kill Ability (PK = 1.0).*** Certain algorithms are based on the assumption of a PK of 1.0. This is a severe restriction and if it cannot be relaxed we caution against use of algorithms of this type.

- ***Static information about various aspects of the battle.*** Many, if not all, formulations assume that the defense is working with static information. The approaches operate on whatever information is available to the battle manager at that moment and often do not take into account possible future events.

- ***Unlimited computing capability.*** Many algorithms have been applied to problems requiring high processing capability. Developers of these algorithms must always keep in mind that algorithms must run in real time and must be highly reliable.

- ***Perfect information about threat parameters.***

c. Figures of Merit

A number of figures of merit for a defensive system are being discussed. Often, they are not compatible; for example, maximum RV destruction and minimum cost. Here we discuss figures of merit for both the strategic defense system and the algorithm used within it to allocate interceptors to targets.

- ***Strategic defense system.*** The several common figures of merit listed below are not to be looked on as representing hard and fast rules. Rather, they should be considered as loose guidelines to those responsible for developing algorithms. For example, the two most commonly used figures of merit are the destruction of as many RVs as possible and the survival of as many value points as possible. The two are not usually compatible. Likewise, if all of the features discussed above are considered in the problem formulation, the algorithm will optimize something which is not obvious, especially if the above features are weighted differently in the cost functions.

A peripheral figure of merit is that the defensive system is robust in all scenarios. This means that the defense should perform well in an interceptor-rich as well as an interceptor-poor environment, should perform well in a high-PK as well as a low-PK environment, and should perform well in spike as well as staggered launches.

- ***Algorithm.*** The obvious figures of merit for any algorithm, assuming that it indeed achieves its objective, are its computation time, the degree to which it can be implemented on specialized computers (such as parallel processors), and how its computation time scales as the problem increases in size.

d. Computational Methods

Several common techniques have been applied repeatedly to the solution of the interceptor-to-target assignment problem.

- ***Shoot at closest target.*** The simplest algorithm is based upon a system in which each CV is totally autonomous and fires at closest targets. This method results in an extremely inefficient use of resources. If the defense is space-based, the assigned intercepts may be kinematically impossible. If the defense is ground-based, interceptors may be expended early and there may be none available late in the battle. This defensive tactic will also result in the same target being fired on by several CVs, with many interceptors wasted. Any concept involving CVs acting in a totally autonomous mode is too easily "spoofed" if the offense launches one missile every 10 to 20 s. Therefore, we reject this "algorithm" as grossly inefficient, as has been repeatedly demonstrated in simulation runs.

- ***Shoot for earliest intercept.*** The shoot-for-earliest-intercept method is representative of what might be used in a system in which each CV is totally autonomous and does not possess global information. It will result in better overall performance than the method described above, but will still result in grossly inefficient use of interceptors and should be rejected as a credible strategy.

- ***Heuristic sorting routines.*** Heuristic sorting methods are based on a coordinated system in which the targets are ranked, possibly according to value or time of intercept. Then, the list is looped through, and the appropriate interceptor is assigned to it, based possibly on time of intercept or on a scheme to distribute the load evenly among CVs. Heuristic routines are implemented in many models. Their main defect is easy "spoofability" and the fact that the user has no idea how much better an optimal-type routine could perform.

- ***Linear programming techniques.*** In linear programming techniques, the classical transportation problem is solved. Applied to a set of SBIs intercepting missiles, the problem is stated as

Maximize

$$\sum_{ij} C_{ij} X_{ij}$$

subject to

$$\sum_{j=1}^{N_T} X_{ij} \leq \text{than the number of SBI on CVi}$$

and

$$\sum_{i=1}^{N_{CV}} X_{ij} \leq b_j$$

where

X_{ij} = number of interceptors to be assigned from CVi to target j,

and

b_j = maximum number of interceptors to be fired at target i.

The user can then decide which items to include in the cost functions (the C_{ij}s). All of the factors discussed above can be incorporated into these cost functions, but typically, in most work seen to date the cost functions include only the number of RVs on a booster (bus) at the time of intercept. It is not obvious which factors are most important and the relative weights to assign to them in developing the cost functions, but they are likely to be scenario-dependent.

Another feature of this problem formulation is that the value of b_{ij} is not universally accepted. The obvious value is 1, but in some formulations it is greater than 1. Any value greater than 1 can result in a waste of interceptors, especially if the PK of the interceptors is high. If the defense is using a high-PK interceptor, this allocation technique will perform well if b_{ij} is set to 1. As discussed below, however, performance degradation may result if the PK is a more reasonable number, say, less than 0.8. This is because only one interceptor per target is allowed in the conventional formulation of this problem. If the defense is interceptor-poor, the linear programming solution is probably adequate in the boost/post-boost phases because the issue of salvoing interceptors should rarely arise.

Several algorithms are available for solving the linear programming problem. The conventional technique has been the simplex technique, described in any textbook on linear programming. More recently, Kharmarkar has developed an algorithm that provides a solution far more quickly. Some contractors are beginning to implement its promising techniques.

- ***Non-linear programming.*** In the case of non-linear programming, the problem is formulated to account for the fact that the interceptor PK is less than unity. For example, if two targets are observed with values of 10 and 1, respectively, the firing strategy to maximize RV destruction for a defense consisting of two interceptors, will be to salvo both at the 10-point target if the PK is less than 0.9. If the PK is greater than 0.9, the better strategy is to fire a single interceptor at each target.

The non-linear programming problem, which will choose the better firing strategy based upon the interceptor PK, can be formulated in a number of ways. One example is for a scenario in which the defense consists of ERIS-like interceptors at N_S sites. There are N_T RVs falling on these N_S sites. The defense might choose to formulate the problem as follows:

Minimize

$$\sum_{j=1}^{N_T} T_j \prod_{i}^{N_s} Q_{ij}^{X_{ij}}$$

subject to $0 \leq X_{ij} \leq m_i$, where m_i = number of interceptors at site i, X_{ij} is an integer, and

$$\sum_{j=1}^{N_T} X_{ij} \leq m_i$$

where

Tj = value of target-j

Qij = (1-Pij), where Pij equals the probability of kill from site-i to target-j

Xij = the number of interceptors allocated from site-i to target-j.

This non-linear problem is more computationally intensive than the linear problem described above. The trade-off between the two will depend on the architecture and weapons. If the defense is interceptor-poor and the interceptor has a high PK, the defense may be able to rely on a linear programming solution and avoid building in the extra computing capability. This trade-off study will have to be performed in greater detail.

3. Review of Algorithm Development Activity

Sponsor: SDIO

Contractor: TRW, Inc.

Algorithm: Farrell-Kharmarkar Approach to Linear Programming

At TRW, the interceptor-to-target-assignment algorithm work is leading toward implementation of a linear programming scheme using a modified-Kharmarkar's algorithm to solve the problem. This scheme should be incorporated into their model "Battle Management Evaluation Tool" (BET) by fall 1987. The algorithm is used within the virtual battle groups, but is not intended as a global solution.

In the TRW concept for battle management, the defense assesses the threat, then forms what TRW calls the virtual battle groups (VBG). These are groups of CVs assigned to certain targets in the threat which are responsible for developing a credible, optimal firing strategy for that battle group and the threat assigned to it. This allows the problem to be formulated on a smaller-than-global level, so that solutions can be obtained quickly.

TRW appears to be in the lead among the five systems architecture contractors in developing algorithms for interceptor-to-target assignment and incorporating them into a weapons and communication system.

Sponsor: IR&D

Contractor: Hughes Aircraft Company

Algorithm: Nonlinear Programming Applied to Late Midcourse Defense

Hughes appears to have made progress in a significant area of interceptor-to-target assignment. They have developed a heuristic, and apparently significantly faster, technique to solve the non-linear programming problem, as it applies to terminal defense. A patent is pending on this algorithm. As soon as it is released we will study its merits and applicability to strategic defense.

Sponsor: OCSA and COTC

Contractor: Alphatech, Inc.

Approach: Novel Solution to the Interceptor-to-Target Assignment Problem

Under this contract, Alphatech undertook a study of a far-term system which includes x-ray lasers. The work was focused on developed coordinated firing strategies in which the interference of the x-ray lasers on the ERIS interceptors was minimized.

Sponsor: SDIO

Contractor: Logicon, Inc.

Approach: Compare and Contrast Several Algorithms

Logicon has investigated heuristic, linear programming, non-linear programming, and dynamic programming approaches to the problem of interceptor-to-target assignment. They favor a linear programming approach, based upon the trade-off between increased system performance and computational costs. They believe that the increased computation time was not worth the slightly better system performance to be achieved by going beyond linear programming. This approach requires a centralized battle manager.

As Logicon sees it, the system will work so that the centralized battle manager will make the "m on n" firing decisions, using the linear programming formulation of the problem. This decision will then be passed off to each CV, where the fire controller resides. As Logicon defines it, the fire controller is responsible for making the detailed decisions regarding fly-outs, while the allocator makes the global decisions. This appears to be a clean separation of the two functions.

Logicon has taken an organized approach to the interceptor-to-target assignment problem. Several techniques were examined, and the trade-offs between performance and computation capability were examined. Logicon is proceeding in an orderly, methodical fashion.

Sponsor: SDIO

Contractor: Martin Marietta

Approach: Heuristic Approach to Redirecting Interceptors in Flight

While Martin Marietta had done little to develop new or novel techniques for the assignment of interceptors to targets, one aspect of their work proved interesting. This is the concept of redirecting an interceptor in flight if the target had been destroyed by an earlier arriving interceptor. Martin Marietta investigated the problem to see if the reallocation concept had merit.

Their conclusion, based on a study in which all interceptors in the vicinity of the redirected interceptor were redirected to other targets, was that reallocation did not contribute to system performance, especially when the added computational cost is considered. This is probably correct. However, the interceptor of interest could be redirected to a booster already targeted by another interceptor, making it the second shot in a salvo. This form of reallocation should have significant merit in a low-PK system, and should tax the system little in terms of added computational loads. Martin Marietta should pursue reallocation along these lines.

Sponsor: U.S. Army Space Defense Command

Contractor: Alphatech, Inc.

Algorithm: Advanced Weapon-Target Assignment

The Advanced Weapon-Target Assignment program involves development, coding, and testing of a set of algorithms which provide solutions to the midcourse and terminal problems. The Alphatech approach involves coordination among SBI, ERIS, HEDI, and even LEDI.

Alphatech began by considering a number of "target-based" weapon-target assignment problems. As described earlier, in this class of problems the defense is usually interested in maximizing the number of RVs destroyed. Because the true problem is NP-complete, Alphatech is pursuing three classes of approaches which provide approximate solutions.

The first class of approaches is one in which integer constraints are relaxed and fractional weapon-to-target assignments are studied. In other words, the results from the algorithms are not constrained to be integers, and thus fractions of interceptors can be assigned. After the fractional assignments are calculated, a subsequent algorithm converts them to integer assignments. The relaxation of the integer constraint allows Alphatech to generate exact solutions to the problem. After examining several algorithms that provided exact solutions to the fractional problem, Alphatech chose to use a primal-dual method based upon monotropic programming techniques. This technique is described in detail in their quarterly report to USASDC, dated April 1987.

The second class of approaches is one in which the constraint of integer assignments is retained in the equations, but a constraint of constant PK is imposed. In other words, while the fractional WTA approach allowed the PK to vary for each interceptor-target pairing, this approach requires a constant PK in order to arrive at a solution. Thus, in cases where one expects little variation among the P_{ij} values, this technique may be appropriate. Alphatech is investigating two different algorithms for solving this problem.

Finally, the third class of approaches being pursued is a maximal marginal return approach, which is a heuristic technique. The idea is, for each additional interceptor, to select from all possible interceptor-target pairings the one which gives the greatest decrease in the expected surviving value of the target. In this area, Alphatech is studying two different algorithms for solution of this problem.

Models of interceptor fly-out profiles and target locations have been developed to support testing of these algorithms, and the algorithms were examined in a fixed scenario to see how they performed as a function of:

- ERIS single-shot PK,
- SBI single-shot PK,
- Inventory of each ERIS farm and CV platform,
- Value structure assigned to the incoming threat, and
- Size of the incoming threat.

The general conclusion was that the fractional algorithm was the most robust in a weapon-rich environment. However, in a target-rich environment (early phases of the SDS), one of the integer-assignment approaches provided more robust defenses. It also turned out that the fractional-assignment algorithm was more computationally intensive than the others. In order to reduce computational time for all algorithms, Alphatech is developing clustering algorithms to reduce the number of targets in their allocation algorithms.

Alphatech is also investigating techniques to maximize CMT survival. This requires a different approach from those described above which maximize RV destruction. This latter approach to the problem can only be used when one has accurate impact point prediction. The complicating feature of this problem is that a single RV can destroy the entire target, and so there is little utility to destroying part of the threat going toward a single target.

Alphatech's approach was to determine which, if any, of the previous techniques could be used to solve this problem. It was found that none was applicable, and two other approaches were taken. One is a maximum marginal return approach, with a pairwise exchange for improvement of intermediate assignments. The second approach was a nonlinear network flow algorithm, followed by a fraction-to-integer conversion algorithm. It should be noted that these algorithms, which attempt to preserve the value of the U.S. assets, are those which would be used for adaptive preferential defense.

Alphatech implemented these algorithms in a scenario and found that they resulted in more sites being saved than was the case with the first set of algorithms, which attempt to maximize RV destruction.

All of the work described above is based upon the defense having firing opportunities at discrete times. Needless to say, the battle manager receives information continuously. Alphatech is investigating approaches that would take into account future intercepts, shrinking footprints, etc. Efforts are also under way to obtain interceptor assignments which minimize the effects of salvage fuzing.

Sponsor: USAF

Contractor: Alphatech, Inc.

Algorithm: Battle-Management Structures

The effort encompasses an integrated set of algorithms for weapon allocation and assignment for the boost and post-boost phases. Both KEW and DEW are considered. The allocation problem is handled by first subdividing the targets into clusters by a zero-one integer programming algorithm. Each cluster is considered a single target in the subsequent allocation algorithm. Weapons are allocated to targets to minimize leakage. Three allocation algorithms were developed and compared for DEW:

- Open-loop linear feedback. This involves a step-wise allocation. First a partial allocation is made, the time to completion of engagement is evaluated for each weapon, and subsequently as many of the remaining targets as possible are allocated. The process is repeated until all opportunities are exhausted.
- Nearest neighbor
- Linear programming.

In the subsequent evaluation, the first was clearly shown to be superior. The weapon-to-target assignment algorithms considered for DEW include variable dwell-time as a function of range and (presumably) aspect angle. The objective is to minimize total dwell-time if there is no deadline to be met. Otherwise, either the leakage or value destroyed is minimized. A number of sequencing schemes were devised to achieve this goal.

For KEW weapon-to-target assignment, the following four techniques were initally considered:

- Linear programming
- Maximum marginal return, as described previously
- Maximum marginal return with a network flow adjustment
- Nonlinear network flow.

Recent work has focused on the iterated linear network programming approach, called ILINE. This is an approximation to the non-linear problem in which successive linear problems are solved iteratively. At each iteration, a maximum of one interceptor is allowed to be assigned to any target. Also, at each iteration only a fraction of the total target set is considered, in order that the higher-value targets (those usually considered) will eventually receive more interceptors than the lower-value targets.

The algorithms represent a solid attempt at a solution to the boost and post-boost assignment problem. Comparing the various candidate methods is particularly worthwhile and the results should be disseminated. They can be carried over to the other phases such as midcourse and terminal. The problem of interaction with attack assessment and selection of the proper algorithm/strategy, or at least the interface with this later problem, needs more attention. Also, the handling of a unified ICBM/ASAT attack needs to be covered.

Sponsor: USAF

Contractor: Logicon, Inc.

Algorithm: Battle-Management Benchmark

The purpose of this study is to develop a capability for comparing battle-management-processor architectures. This is being accomplished in a number of steps:

- A test-case SDI architecture was defined; it consists of a space-based architecture for boost-phase engagement
- Response time-lines were evaluated; these yielded the algorithms' execution speed requirements

- Three stressful battle-management functions were defined, namely, (a) preplanning/replanning weapon-to-target space allocation; this function allocates weapons to a target space segment through which targets might fly, (b) options are generated and defined, each associated with a particular attack scenario, and (c) option selection and execution of weapon-to-target assignment.

The effort regarded in its entirety shows an impressive unified approach which could be applied with great benefit to other portions of the algorithm-development program. The scope of the effort is much wider than it seems at a first glance; for example, the total number of algorithms being developed is large, and may be categorized as follows:

- Preplanning/replanning space allocation

  --Bellman's dynamic programming algorithm
  --Larson's dynamic programming algorithm
  --Dantzig's linear programming algorithm
  --Polynomial-time (Karmarkar's) linear-programming algorithm
  --Network linear programming algorithm

- Two weapon-to-target-assignment opportunities algorithms are used in the above, one each for KEW and DEW. Each uses the relevant parameters, e.g., line of sight, range, velocities, etc., to compute shoot opportunities.

- Option selection

  --Network linear programming algorithm.

The basic framework is in place for evaluating the computational and storage requirements of the algorithms for various selected scenarios; in fact, some preliminary results have already been obtained.

G. MODE SELECTION

1. Description

Mode selection for strategic defense appears to be one of the least studied issues. The consequences of the defense's mode, or readiness level, are many and will affect overall system performance. Some of these consequences are to be discussed here.

Because system performance will depend on the readiness level, we recommend that the SDIO begin a study on this subject. The "products" of the study should be a definition of the various readiness levels, how they are set, and their effects on the system. Once this is done, all work related to the parts of the system affected by the mode level, e.g., the weapons subsystems, must be made to conform to the mode level, rather than be mode-independent, as is currently the case. Primarily, this work will apply to the testbeds.

The extent to which humans will interact with the system might depend on the readiness level. For example, in times of crisis, humans may be removed from the loop and the system set to operate in an automatic mode. At lower readiness levels, however, humans might be required to enable weapons.

The relative values of the U.S. targets might change as a function of readiness. For example, at a high level of readiness, submarine bases may be reduced in value if the submarines are not in port. Likewise, bomber bases may be reduced in value if the planes are flushed. Also, the relative values of population centers and missile silos might change with different readiness levels.

The nature of the defense may also depend upon the level of readiness. For example, in peacetime, if the defense sees a modest launch of missiles, it may assume these had been launched by mistake and salvo at them to insure their destruction in the boost phase. However, in times of heightened tensions, the defense may choose to hold weapons back for a subsequent wave of missiles.

2. Critical Features

a. Factors

Several factors may be used in defining the level of readiness. We expect some to be the same as those used to define the DEFCON levels. Inputs to the decision process will include intelligence reports, the existing political climate, and any prior or current military activities under observation. We strongly recommend that the existing rules for determining readiness levels be reviewed when developing new rules for the defense system.

b. Assumptions

Some underlying assumptions might be made when developing the rules for readiness levels. First, it is assumed that the timelines of the battle will allow the participation of man in the loop if a certain readiness level requires human activation of the system. Second, the assumption could be made that notice of a change in mode selection can be propagated to all battle managers within the time needed to use this information.

Third, it might be assumed that the battle-management algorithms are sophisticated enough to respond in different ways to a given threat, depending on the readiness level.

c. Figures of Merit

Any algorithm developed for mode selection must be flexible enough to respond to and direct the defense in a wide range of scenarios. This flexibility, and its resistance to failure under a new scenario, should serve to characterize the "goodness" of the output of the mode selection process.

3. Review of Algorithm Development Activities

Sponsor: Army

Contractor: Sparta

Approach: Battle Planning (BP)

The goals of the BP algorithm are to adaptively select strategy based on threat, defense, and asset status, and to destructure structured attacks. Seven specific tasks are outlined with detail the subfunctions of the BP algorithm.

(1) Adjust Threat Assessment (TA) threshold

(2) Adjust mid-course sensor threshold for each sensor class for discrimination of each of the nine object-type models

(3) Identify targets as candidates for engagement planning

(4) Provide weapon availability and strategy to engagement planning

(5) Provide target list to engagement planning based on the value of the threat and battle-performance requirements

(6) Evaluate battle planning progress

(7) Adjust sensor-management regions and boundaries.

The briefing materials reviewed indicate that much has gone into this algorithm to determine precisely what information it needs from, and must provide to, the overall battle-management network. However, no information was available about specific strategies, how they are selected, and how the goals for these strategies are to be achieved. No analysis was shown of the optimization technique for sector boundary selection, but these details can be worked out during the remainder of this program.

H. KILL ASSESSMENT

1. Description

It is not clear which, if any, algorithms would be needed for kill assessment, and adequately little effort has been devoted to determining exactly how it is to be performed. There are several possibilities. The defense could choose to observe the object under fire and note any change in its velocity if the kill is to be by a KEW. Or, a beacon could be placed on the KEW which would stop broadcasting in the event of a successful intercept. A third possibility would be to observe the object some time after the intercept to see if it is at the position predicted on the basis of tracking data. Since the function of kill assessment has not been fully defined, it is strongly recommended that the SDIO provide some guidance on how the kill assessment function is to be performed, so that contractors can begin to develop algorithms and incorporate them into the BM/C^3 architecture.

2. Features

a. Critical Factors

The major factors which will influence the kill assessment function and the choice of how it is to be performed are those relating to sensors. The nature of the kill mechanism, whether applied to a hard or soft kill, will determine the type of sensors needed, and in the case of a soft kill, whether or not the function can even be performed. Background clutter will be a constraining factor, probably requiring sensors to look above the horizon.

b. Assumptions

The major assumption is that the kill makes a noticeable change in the object. For a soft kill, this may not occur.

c. Figures of Merit

For any kill-assessment algorithm, there are two figures of merit. First is the length of time required to perform the function, which in turn will depend on how it is performed. If done by a sensor, the sensor update rate will be the driver. The second figure of merit is how reliable the kill assessment data is. Obviously, the data will not be perfect, but if this function cannot be performed with a high degree of confidence, the defense will suffer degraded performance. If this function cannot be performed well, the other choice is to design a system to operate without it.

Sponsor: Army

Contractor: Sparta

Effort: Engagement Asset (EA)

The goal of the EA function is to account for engagement failures using information available either explicitly or implicitly, while minimizing delay in reporting these failures. Subfunctions to engagement assessment are the following:

- Receive designation advisory from Local Sensor Resource Management (LSRM)
- Assess engagement process, update EA database, and send engagement status to Battle Assessment (BA)
- Update engagement assessment criteria as received from Battle Planner (BP).

The engagement asset function basically performs what is called kill assessment. The reviewed material does not provide example criteria for assessing an engagement,nor does it describe the format (measurements, probabilities, etc.) of the designation advisory. The engagement-assessment criteria comes under task 6 of the Battle Planner (BP), but is presently only a functional model. The partitioning of the assessment process between the LSRM and the EA must provide some payoff in decreased communications load, but it is unclear whether the entire assessment process could be performed on the lead satellite under the LSRM.

I. SENSOR RESOURCE MANAGEMENT

1. Description

The purpose of the sensor-resource-management algorithms is to establish multi-sensor fusion nodes and volume surveillance responsibility. These algorithms primarily apply to the AOS, SSTS, and perhaps the probe sensor systems, since the complex relative motions between the sensors and the threat will require a dynamic configuration process, rather than fixing a platform's surveillance volume, as might be done on BSTS. As described in Section III.B, "Track Initiation and Maintenance Algorithms," several sensor platforms, all scanning the same volume of space, may report their tracks to a central node (generally on one of the sensor platforms) where the tracks are fused to increase their accuracy. The output of a sensor-resource-management algorithm would be an association-vs.-time matrix showing (1) which sensor platforms are coordinated, (2) which one is acting as the fusion node, (3) how long the configuration would be maintained, and (4) perhaps estimates on successive configurations so that pending communications and surveillance problems could be identified in advance.

2. Critical Features

a. Factors

1. ***Orthogonal viewing angles.*** The primary benefits from multisensor fusion are derived by combining observations of a set of targets from multiple viewing angles. Therefore, a group of three coordinated sensors which provide observations from three orthogonal views (targets at the origin with a sensor along each axis) will provide better sensor fusion than if the sensors and the threat are all coplanar, or even worse, close to colinear.

2. ***Traffic load.*** Assignment of surveillance volumes should not be such that the number of objects that a given sensor platform is responsible for tracking exceeds that platform's computational or memory resources.

3. ***Configuration stability.*** During the changeover from one configuration to the next, some track handover between sensors moving out of the surveillance area and sensors moving into it will be required. Since this process will take some time, and the handover will place demands on the communications network, it would be desirable that a sensor configuration last as long as possible.

4. ***Robustness and survivability.*** Surveillance responsibility and track fusion nodes should be so distributed that the loss of any platform does not interrupt the surveillance and tracking function; e.g., loss of a single platform serving as a fusion node for two groups of sensors and itself responsible for a dense volume of space would make the system vulnerable.

b. Assumptions

Due to the small level of effort we have seen in this area, a list of commonly made assumptions cannot be made at this time.

c. Figures of Merit

1. ***Sensor-threat orthogonality.*** Maximizing the orthogonal viewing components between a set of coordinated sensors and a surveillance volume is one method of selecting those sensor platforms to be grouped. See "Orthogonal viewing angles" above.

2. ***Traffic loading.*** Traffic loading means selecting groups of sensors and their associated surveillance volumes so that the computational load associated with the tracking and discrimination functions is distributed among the sensor platforms as evenly as possible. See "Traffic load" above.

3. ***Configuration stability.*** Selecting a group of sensors so that the time over which those sensors can monitor their assigned surveillance volume is maximized is referred to as configuration stability. See "Configuration stability" above.

d. Computational Methods

1. ***Linear programming.*** Linear programming techniques could be used to select either surveillance volumes or the sensor platforms to monitor them. The cost function could be some combination of the metrics listed above.

2. ***Heuristic methods.*** Depending on how sensitive tracking performance is to the factors listed above, heuristic methods may suffice to handle the sensor-management problem.

3. Review of Algorithm Development Activity

Although the function of sensor platform grouping and coordination (so-called "battle groups" without the weapons) is widely discussed, we have seen only one detailed example of work in this area. Many of the battle-management testbeds being planned would be extremely useful in executing sensitivity studies of tracking performance, discrimination performance, and defense robustness to ASATs with respect to the factors listed above. Since the outputs of this function are greatly dependent on the threat, detailed Red Team scenarios for simultaneous launches, staggered launches, and combined ASAT/ICBM launches are necessary for full evaluation of these algorithms.

Sponsor: Army

Contractor: Unisys (System Development Corporation -SDC)

Algorithm: Regional Sensor Resource Manager (RSRM)

The goal of the RSRM algorithm is to optimally form trios of surveillance satellites (SSSs) which cover a common volume of space for sensor-to-sensor correlation. A lead sensor is defined which performs the sensor-to-sensor correlation function, and two associate sensors. Individual sensors are free to serve as members of more than one sensor trio.

Sensor selection is based on maximizing a weighted sum of three metrics:

- Time-sampled LOS orthogonality of sensors with respect to center of sector
- Combined time-sampled percentage of sector volume within field of view of all three sensors
- Use of surveillance platforms.

The orthogonality metric is useful not only for the tracking function, but also for the discrimination function (orthogonal viewing/imaging aspects).

The detailed description of sensor selection shows a great deal of work. The only remaining detail would be selection of the weights for the three metrics. Since similar parameters (thresholds, etc.) are determined and passed by the higher level planning functions, these weights might be determined by either the Battle Planner (BP) or Battle Assessment (BA) algorithms (of the Army's Algorithm Development Program, ADP).

IV. REVIEW OF ALGORITHM DEVELOPMENT PROGRAMS

This chapter provides summaries on the algorithm development programs organized by the services, the SDIO, and the special project offices (BSTS, SSTS, and SABIR) whose individual algorithms are discussed in detail in Chapter III.

A. THE SDIO BM/C^3 OFFICE THROUGH THE SERVICES

1. Army[3]

The three algorithm development activities sponsored by the Army which we reviewed are:

- The Army's Algorithm Development Program (ADP)--Unisys and other contractors
- Multiple Information Set Tracking Correlator (MISTC)--Alphatech Inc. and Honeywell
- Advanced Weapon-Target Assignment Algorithms--Alphatech Inc.

The ADP was intended to provide a complete end-to-end set of battle management algorithms for strategic defense. The architecture used in their design was primarily a ground-based architecture, with some space-based sensors such as SSTS. ERIS and HEDI were the only weapon systems used; no boost-phase defense or BSTS was included in their architecture. The charts on the following pages summarize and critique the individual algorithms in the ADP, as well as the other two algorithms listed above. Detailed summaries on these algorithms appear in Chapter 3.

In general, the Army's ADP has done an excellent job of defining most of the algorithms necessary for a ground-based defense, and how they connect with one another.

---

3 As this report is being issued, the Army is in the process of updating its architectural concepts and the associated algorithms, in order to bring it in harmony with the latest SDIO architecture. The most important change is the addition of space-based interceptor platforms. Lack of time does not allow inclusion of these updates.

There is considerable original work here, and they have addressed algorithm coordination issues which we have not seen elsewhere. However, many of the more complicated algorithms (such as Engagement Planning) remain "empty boxes", with only inputs and outputs well defined. However, methods for deriving those outputs have not yet been determined. Other algorithms need additional development (the discrimination algorithm needs to include threat parameter estimation), but on the whole, the ADP represents a coordinated, well-thought-out effort.

The coordination between the many contractors working on the ADP is a key issue. It was obvious that there was "cross fertilization" between the algorithm development teams. Information produced as a side effect in the execution of one algorithm was often put to creative use in another algorithm in areas such as threshold setting and determining a set of weights in a cost function.

2. Air Force

The charts on the following pages summarize and critique the four algorithm programs we reviewed that are being sponsored by the Air Force. Detailed summaries appear in Chapter 3. The Air Force lacks a coordinated program for developing an end-to-end algorithm set, depriving them of the advantages such a program would provide.

However, the individual algorithms show some innovative and promising approaches to the boost-phase tracking and weapon-to-target assignment problems. Applied Technology Associates (ATA) is doing some novel work in template matching for tracking in the boost phase, and PAR Technology Corporation rather than forming 3-D tracks on each sensor platform, all correlation and track formation is done at a central node where focal plane information from each satellite in a group is passed. Although this places a greater communications load on the system, we have not yet seen any sensitivity studies comparing algorithms of this type to others using the "form tracks on-board the sensors" philosophy. Such results could prove useful to later architecture studies.

The weapon-to-target assignment algorithms being developed by Logicon and Alphatech show a wide scope in that they attempt a numerical comparison of the performance of various algorithms.

ARMY SDI BATTLE MANAGEMENT ALGORITHMS

Algorithm/Contractor	Status	Summary	Critique
Regional Sensor Resource Manager (RSRM)/Unisys (System Development Corporation -SDC)	Functioning on Army's DMC/ARC	• Optimally forms trios of surveillance satellites to cover a common volume of space • Performs data base housekeeping	• Metrics for optimization well suited for both tracking and discrimination purposes • Only algorithm reviewed which describes sensor group formation in detail • Weights for metrics are static; ADP design philosophy might have these determined by the BP or BA algorithms
Local Sensor Resource Manager (LSRM)/ General Research Corporation (GRC)	Functional design; demonstration in December 1987	• Organizes single-sensor tracks for multi-sensor correlation • Supervises track handovers • Performs guidance updates and target designations • Performs local threat assessment	• None of the methods or approaches for any of the tasks listed were described in the available materials • Appears to be molded specifically for DMC/ARC testbed, rather than an actual scenario
Multi-Sensor Correlation and Target State Estimation (MSCTSE)/Nichols Research Corporation (NRC)	Functional design; demonstration in September 1987	• Provides sensor-to-sensor correlation of single-sensor generated tracks • Assignment approach includes multiple observation combination filters to reduce computation time	• Process described for surveillance satellites only (not for AOSs or GBRs) • Design philosophy driven by reducing communications and processing requirements • Process for setting filter thresholds not described • Scan-to-scan correlation problem and track initiation not described; left as a sensor problem
Multiple Information Set Tracking Correlator (MISTC)/ Alphatech and Honeywell		• Coordinated algorithm set for forming tracks, fusing tracks, and maintenance of tracking nodes • Scan-to-scan correlated tracks formed on sensor	• Has provided valuable insights into the global track formation and management problem by deriving data structure, a hierarchical sensor architecture, and a method of asynchronous communications

(continued)

ARMY SDI BATTLE MANAGEMENT ALGORITHMS (continued)

Algorithm/Contractor	Status	Summary	Critique
Multi-Sensor Discrimination and Classification (MSDC)/Nichols Research Corporation (NRC)	Functional design; demonstration in September 1987	• Static weights used for combining LWIR and LADAR imaging data • Nine probabilities assigned to each object at sensor level (3 junk, 3 decoy, and 3 RV types) • Probabilities fused into nine correlated probabilities	• Other ADP algorithms could provide valuable information for dynamic weights for observables • Heavily dependent on *a priori* threat models; no estimation performed • With only probabilities passed from the sensors, valuable discrimination information (multiple viewing aspects) may be lost
Advanced Weapon-Target Assignment Algorithms/Alphatech		• Fractional and integer weapon-to-target assignment algorithms being developed • Methods to maximize marginal return of individual interceptors	• One of the few assignment algorithms in survey to include salvoing • Approaches being analyzed are important, but do not constitute a complete set of algorithms for solving the assignment problem • Shoot-look-shoot with same or dissimilar weapons (ERIS, HEDI, LEDI) not included
Battle Planning (BP)/ Sparta	Functional design; demonstration in March 1988	• Adaptively selects strategy based on threat, defense, and asset status • Strategies to include methods for destructuring attacks	• Good connectivity to other algorithms in the Army ADP • No information available on what the strategies are, how they are selected, and how the goals for these strategies are to be achieved
Engagement Planner (EP) Unisys (System Development Corporation - SDC)	Functional design; demonstration in December 1987	• Optimally performs weapon-to-target assignment task • Allocation is a function of strategy, asset status, and threat status as provided by the other algorithms of the ADP	• Unique ideas include preferential defense to ensure minimal survival of several classifications of targets, rather than maximizing survival points • No material was available to describe the approach to the problem or how the many inputs are to be used

(continued)

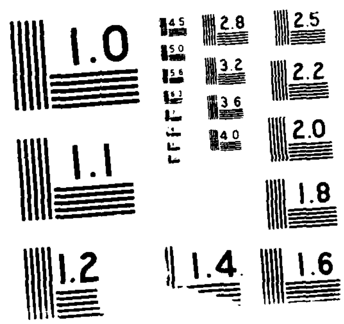
ARMY SDI BATTLE MANAGEMENT ALGORITHMS (continued)

Algorithm/Contractor	Status	Summary	Critique
Threat Assessment (TA)/ Sparta	Functioning on Army's DMC/ARC	• Recognizes threat structure • Estimates types of damage for each threatening object tracked • Provisions for time-dependent defense strategies	• Only algorithm reviewed which is concerned with threat structure • Impact and damage estimates may be too detailed for the quality of discrimination available
Engagement Assessment (EA)/ Sparta	Just coming online on Army's DMC/ARC June 1987	• Accounts for engagement failures • Performs database housekeeping between the Local Sensor Resource Manager and the Battle Assessment and Battle Planner algorithms	• Assessment function is actually split between the LSRM and this algorithm • Material reviewed does not describe assessment criteria
Battle Assessment (BA)/ Sparta	Just coming online on Army's DMC/ARC June 1987	• Provides mechanisms to account for engagement failures • Summarizes and computes engagement statistics • Performs data base housekeeping • Monitors and reports on interceptor status	• Material reviewed does not describe mechanisms to account for engagement failures • Data link between interceptors and BA algorithm listed as part of DMC/ARC driver routine, and is not simulated as part of a physical system
Global Information Manager (GIM)/ Unisys (System Development Corp. - SDC)	Functional design; demonstration in March 1988	• Collects and maintains status data • Detects and confirms events • Indicates problems in links and data bases • Maintains information consistency	• Important algorithm--only one of its kind reviewed • Some subfunctions and trade-offs incomplete, but outlined for future work

AIR FORCE SDI BATTLE MANAGEMENT ALGORITHMS

Algorithm/Contractor	Status	Summary	Critique
Tracker-Correlator/ Applied Technology Associates (ATA)	To be included	• Single sensor track initiation in boost phase • Single and multiple sensor track maintenance • Single sensor/single object only completed to date • Two-scan initiation in early boost phase by fitting observations to nominal booster flyout profile	• Novel track initiation approach--very useful for track initiation in early boost • Issue of scan-to-scan correlation in first two observations in multi-object scenario will be addressed in follow-on effort
Boost Phase Analysis (track initiation and maintenance)/PAR Technology Corporation	To be included	• Boost phase track initiation and maintenance • Raw data passed from sensors to battle manager • 3-D correlated tracks formed at fusion satellite/BM node	• Only tracking algorithm in survey to form tracks from multiple sensors, not at each sensor • Performance and communications load of this approach have not fully been determined
BM Benchmark Algorithms/Logicon	To be included	• Study to develop capability for comparing BM processor architectures • Space-based boost intercept architecture • Algorithm speed requirements set by scenario time lines • Five allocation, three assignment, and one option-selection algorithm are compared	• Very useful framework for evaluating performance, computational requirements, and storage requirements of a loose set of algorithms for various selected scenarios
Battle management structures/Alphatech	To be included	• Three allocation algorithms were investigated; their outputs are used by the subsequent assignment algorithms • KEW and DEW assignment with clustering of targets to reduce scale of assignment problem • Multiple approaches for DEW sequencing being analyzed; comparative data has been produced • KEW assignment approaches include salvoing	• Comprehensive set for midcourse engagement; can be carried over to other phases • Comparisons being made are very useful and should be disseminated • Algorithms not designed to interact with strategy section and attack assessment algorithms; this area needs more attention

AD-A194 357 SDI (STRATEGIC DEFENSE INITIATIVE) BATTLE MANAGEMENT/C3 2/2
(COMMAND CONTROLAA (U) INSTITUTE FOR DEFENSE ANALYSES
ALEXANDRIA VA G FRENKEL ET AL APR 88 IDA-P-2068
UNCLASSIFIED IDA/HQ-87-32185 MDA903-84-C-0031 F/G 15/3.1 NL


1.0
1.1
1.2
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

NAVY SDI BATTLE MANAGEMENT ALGORITHMS

Algorithm/Contractor	Status	Summary	Critique
Contact Discrimination Algorithm/ Command Systems Group, Inc. (CSG)	Code functioning; test results available	• AI techniques to fuse LWIR and LADAR observables • Uses imaging during RV and penaid deployment • Uses object characterization and payload partitioning expert systems	• Very detailed, comprehensive approach to discrimination problems • Not concerned with communication loads associated with coordinating raw observable data from multiple sensors • May be too dependent on deployment imaging • May not be robust to errors in *a priori* observable models; sensitivity studies needed
Tracker-Correlator/ VERAC, Inc.	Exact status unknown; presumed to be beyond functional model	• Separates tracks from sensor reports into clusters, each having a dedicated local processor • Target discrimination-type observables used in correlation process • Track "Scene" tree generation and pruning for each cluster	
Situation Assessment/ Strategic Planning/ Gould, Inc.	Functional model only		
Weapon Allocation and RF Spectrum Management/AT&T	Functional model only		

3. Navy

The Navy's algorithm development program appears to be far behind those of the Army and Air Force in both schedule and coordination. At this time there is not enough information to make an overall assessment of the Navy program.

B. THE SDIO BM/C^3 OFFICE THROUGH CONTRACTORS

There are several algorithm efforts being funded by the BM/C^3 office of the SDIO directly, most of which deal with the problem of weapon-to-target assignment as shown in Figure S-1. IDA has also developed several algorithms in the areas of resource basing and discrimination. The motivation for this work resulted from our recognition of several gaps in the overall state of algorithm development which we have noticed from this survey of the contractor's efforts.

C. THE SDIO THROUGH ELEMENT PROGRAMS

1. BSTS Special Project Office

As part of the BSTS development program many algorithms are being developed. During the information gathering phase of this activity, we were "read" into this program, which has Special Access Reqirements (SAR), and subsequently given a briefing on the top-level BSTS system concept. Further information on the details of the various algorithms was denied on the basis of a lack of need-to-know. It is our opinion that information about BSTS which is badly needed by the SDI community is not available, to the detriment of the overall technical effort. A careful reevaluation of material for classification purposes would be beneficial at this point.

2. SSTS

Due to time constraints and problems with scheduling meetings with the contractors, a complete list of the algorithms being developed under the SSTS program is not contained in this report. However, notes from early discussions with the people at Hughes Aircraft Co. who are developing tracking algorithms for SSTS are included in Chapter III.

3. Space-Based Interceptor Program

We visited the Program Office and were given a briefing. In addition, relevant reports were obtained. Algorithms were developed by the SDI program for the intercept end-game, and since these are not BM/C^3 algorithms, they were not included in this report. The only BM/C^3 algorithms developed are for weapon-to-target assignment by Logicon and Martin Marietta, and these are described in Chapter III.

D. THE SDIO OUTSIDE THE BM/C^3 PROGRAM

A number of models have been developed for the Strategic Architectures Directorate. Each of the five contractors in Phases I and II has in-house models. Likewise, the SDIO has funded development of two other models. Our opinion is that most of these models are simple design tools, whose purpose is to test candidate architectures. They are not algorithm testbeds, nor do they contain innovative algorithms. The only model which was designed to serve as a testbed is TRW's model, BET. This is a model designed to study different algorithms within the TRW BM/C^3 concept. A listing of the companies and associated models is given below.

Martin Marietta	Model Toolset for Evaluation of Strategic Architectures (MESA)
Rockwell	End-to-End Engineering Model (ETEEM)
SAIC	Strategic Defense System Performance Evaluator (SDS-PA)
SPARTA	Defense in Depth Simulation (DIDSIM)
TRW	Battle Mangement Evaluation Tool (BET)
Blime, Inc.	Strategic Offense and Defense Simulator (SODSIM)
ANSER, Inc.	Blue Defender.

The Innovative Science and Technology Program initiated an algorithm development activity which will be monitored for future results.

E. ALGORITHM DEVELOPMENT OUTSIDE THE SDIO

There are some algorithm development programs, such as IR&D and algorithms developed prior to SDIO, which are of interest. Three of these--MVADEM for resource basing, MITRE for track initiation and maintenance, and Hughes for weapon-to-target assignment--are described in Chapter III.

END
DATED
FILM
8-88
DTIC